MIKROSZÁMÍTÓGÉP MAGAZIN

A NEUMANN JÁNOS SZÁMÍTÓGÉP-TUDOMÁNYI TÁRSASÁG LAPJA

1984/2

VIDEOTON

ELEKTRONIKAI VÁLLALAT

16 és 8-bites feldolgozás,
MS/DOS, CP/M 86 és CP/M 3.0
kompatibilis operációs rendszerek,
szoftver kompatibilitás IBM személyi számítógéppel,
kiváló minőségű grafikus ábrázolás,
nagy operatív és háttértár kapacitás,
távadatfeldolgozás.

VT 16 személyi számító-gép

MIKROSZÁMÍTÓGÉP MAGAZIN

A Neumann János Számítógéptudományi Társaság lapja

A kiadvány a Tudományszervezési és Informatikai Intézettel együttmûködve készül

A szerkesztő bizottság vezetője: Kovács Győző

Munkatársak:

Broczkó Péter (hírek)
Buday György István (személyi számítógépek)
Garádi János (agyafúrmány, rövid és ravasz programok)
Jakab Ágnes (ember–gép kapcsolat)
Kovács Győző (levelezés)
Lindner László (sakkprogramozás)
Nacsa Sándor (termékismertető)
Pataki Ernő (programozástechnika)
Petróczy Judit (könyvek)
Pogány Csaba (alkalmazástechnika, tanfolyam)
Simonyi Endre (klub)
Varga András (iskola – számítógép)
Vass Nándor (alkalmazások)
Votisky Zsuzsa (játékprogramok)

Felelős szerkesztő: Könyves Tóth Pál

Szerkesztőség: Budapest V., Báthori u. 16. Telefon: 329-349, 329-390

Kiadja: a Lapkiadó Vállalat
Felelős kiadó: Siklósi Norbert vezérigazgató
Kiadóhivatal: Budapest VII., Lenin krt. 9–11. Postacím: 1906 Budapest, pf. 223. Telefon: 429-350, 221-285

Terjeszti a Magyar Posta
Előfizethető bármely postahivatalban, a kézbesítőknél, a Posta hírlapüzleteiben és a Posta Központi Hírlap Irodában (Budapest V., József nádor tér 1., postacím: 1900 Budapest) közvetlenül vagy postautalványon, valamint átutalással a PKHI 215-96162 pénzforgalmi jelzőszámra.
Előfizetési díj: egy évre 168,– Ft, fél évre 84,– Ft.

Szedte: a Nyomdaipari Fényszedő Üzem (847362/09)

Nyomás: Petőfi Nyomda, Kecskemét, Külső Szegedi út 6. (84.41778) Telefon: 20466
Felelős vezető: Ablaka István igazgató

INDEX: 25 629

Címképünk: a Számítástechnikai és Ügyvitelszervező Vállalat szolgáltatásai

Tartalom

Együtt az Olvasókkal 2
Adok – veszek – cserélek 36
Ismerkedjünk a számítógépes grafikával! 40
A játékprogram-pályázat eredménye 45

SZEMÉLYI SZÁMÍTÓGÉPEK

Mi a személyi számítógép? II. 3
Véleményem szerint 4

ISKOLA – SZÁMÍTÓGÉP

Számítástechnika az oktatásban 6
Csapdák az RND körül 8
Oktatási programajánlat 9
Személyi számítógépek a matematika oktatásában 10
Tanulja meg minden hallgató! 11
A „10 000 mikroszámítógép" program 13

TANFOLYAM

Alapozás III. 14

PROGRAMOZÁSTECHNIKA

Mit tud az APL programnyelv? 16
BASIC interpreterek – a tár szervezése 17

TERMÉKISMERTETŐ

VU–CALC elektronikus feladatlap ZX–Spectrumra 21
A Floppymat család 23

SZÁZLÁBÚ 24

EMBER–GÉP KAPCSOLAT

Terminálok – a felhasználó szemszögéből 29

μKLUB

Karaktersorozat-keresés BASIC-ben 29
ZX81-tulajdonosok, figyelem! 30
Építsünk számítógépet! I. 30
Válasz Katona László levelére 31

JÁTÉKPROGRAMOK

Szópóker (1 k-s ZX81 gépekhez) 32
Csillag (16 k-s ZX81 gépekhez) 32
Kígyós játék (HT–1080Z iskolaszámítógéphez) 33
A videojáték-pályázat eddigi eredményei 33

RÖVID ÉS RAVASZ PROGRAMOK

Új kaleidoszkóp (HT–1080Z iskolaszámítógépre) 36
Milyen napon születtem? 36

AGYAFÚRMÁNY 37

AZ OLVASÓ ÍRJA 38

KÖNYVEK 44

HÍREK, ÉRDEKESSÉGEK 45

SAKKPROGRAMOZÁS

A 3. Mikroszámítógép Sakkvilágbajnokság 46

Együtt az Olvasókkal

„Gyorsuljatok, vonatok, repülők,
kicsinyüljön a föld,
s nőjön, mely szemünk láttán áll elő,
az a sorsunkat is kézbe vevő
emberi, új erő!"
(Illyés Gyula: Óda a gyorsasághoz)

Százharmincöt kérdőívet küldtek vissza olvasóink február közepéig lapunkra vonatkozó véleményekkel. Ezenkívül még százan írtak levelet és küldtek jó tanácsot, programokat, ötleteket, rajzokat, és szinte mindenki gratulációt és jókívánságokat. Köszönöm a szerkesztőség nevében.

Persze kritikus hang is volt, de disszonáns alig, talán kettő. Ebből is tanulunk.

A feldolgozott kérdőívek gyakorlatilag az 1983. évi számunkat értékelték, így az alábbi táblázat erre vonatkozik.

KÉRDÉSEINK	IGEN	NEM
Számítástechnikus?	42	93
Talált érdekes cikket?	125	0
Sokat	50	–
Eleget	70	–
Keveset	11	–
Egyet sem	0	–
Jó-e a rovatszerkezet?	110	9
Írna-e a μ M-ba?	78	34
Jó-e a lap neve?	118	15
Tetszik-e a lap formája, tördelése?	107	15
Jónak minősíti-e az egyes rovatokat?	59	0
Iskola – számítógép	45	14
Tanfolyam	68	12
Programozástechnika	96	4
Termékismertető	63	13
Piac	53	14
Ember–gép kapcsolat	43	19
μ Klub	66	3
Agyafúrmány	60	7
Az olvasó írja	42	11
Játékprogramok	95	4
Rövid és ravasz programok	79	7
Hírek, érdekességek	74	6
Új könyvek	64	7
Adok, veszek, cserélek	65	3
Sakkprogramozás	43	16
Szoftverpiac	67	2

Meg kell jegyeznünk, hogy valamennyi beküldő csak az első kérdésre felelt, nevezetesen: számítástechnikus szakembernek tartja-e magát. A többi kérdésre volt aki felelt, volt aki nem. A táblázatban ezért nem kapott valamennyi sor 135 „szavazatot".

Az eredmény azt mutatja, amit vártunk és reméltünk (ha ez a válasz-tömeg jó mintavételnek tekinthető), hogy lapunk olvasói inkább nem számítástechnikai szakemberek. Ennek az aránynak nagyon örülünk. Névtelen levelet nyolcat kaptunk, ebben volt az a kettő is…

VÁLASZOLÓ OLVASÓINK FOGLALKOZÁS SZERINTI ÖSSZETÉTELE:	
Egyetemet, főiskolát végzett (közülük 2 nyugdíjas)	72
Közép- és általános iskolai tanuló	25
Középfokú végzettségű számítástechnikai munkatárs	12
Egyetemi és főiskolai hallgató	9
Szakmunkás	8
Sorkatona	1
Ismeretlen foglalkozású	8

A legtöbb választ küldő, diplomás olvasóink foglalkozása: középiskolai tanár 19, villamosmérnök 17, mérnök (szakmegjelölés nélkül) 9, programozó 4, vegyész, vegyészmérnök 4, közgazdász 4, rendszermérnök 3, főiskolai tanár 3, gépészmérnök 3, fizikus 2, szociológus 1, matematikus 1, közlekedési mérnök 1, orvos 1.

A választ küldők közül 78-an írnának a μ M-ba, nagyon sokan kértek írásaikhoz ötleteket. A legegyszerűbb talán, ha felsoroljuk az olvasói kívánságokat, hiszen ezeknek a lehetőségeken belüli teljesítése a szerkesztőség részére kötelező.

Olvasóink közül sokan szeretnének számítógépet építeni, illetve meglévő számítógépeiket (HT–1080Z, Sinclair, Commodore) átalakítani vagy kiegészíteni. Olyan sorozatot kérnek, amelynek alapján az egészen kezdők is meg tudják építeni saját számítógépüket. Ezt a sorozatot már meghirdettük, ha sikerül, akkor a gép építéséhez szükséges alkatrészeket egy csomagban (kit-ben) is meg lehet majd vásárolni.

Többen kérték a HT, illetve Sinclair gépek ROM leírását, aztán hardver ötleteket, például a gépek memóriájának bővítésére, különleges perifériák hozzákapcsolására, sőt olyan kívánság is érkezett, hogy foglalkozzunk azzal, hogyan lehet ezeket a házi számítógépeket telefonvonalon nagyobb számítógépekhez kapcsolni.

Olvasóink igen nagy része valószínűleg programozni szeret, legalábbis a kívánságlistát tekintve erre lehet következtetni. Legnagyobb meglepetésemre nem is magas szintű programozási nyelven, például BASIC-ben szeretnének programokat írni, hanem a gépi kódú programozást szeretnék megtanulni. (Tulajdonképpen érthető, otthoni, iskolai számítógépük memóriája viszonylag kicsi, és kis helyen elférő, sokat tudó programot csak gépi kódban lehet írni!)

Jogos kívánság, csak nem tudom, hogy teljesíteni tudjuk-e, hogy közöljünk módszereket (technológiát?), hogyan lehet portábilis, azaz egyik géptípusról a másikra könnyen átvihető programokat készíteni. Elhangzott olyan kívánság is, hogy foglaljuk össze a BASIC „nyelvjárásokat" egy cikksorozatban; ezzel is megkönnyíthetjük a programcserét. Talán a legfőbb kívánság, hogy írjuk meg, Sinclair, ABC–80, Commodore programokat hogyan lehet könnyen futtatni az iskolaszámítógépen.

Szerzőket keresünk olyan cikkek írására, mint például matematikai, fizikai feladatok, kulcsszó-felismerés algoritmusra, matematikai függvények ábrázolása, képernyőkezelési trükkök.

Várunk ötleteket, amelyeket vagy pályázat, vagy feladat formájában adunk közre, és ha valaki elkészíti a programot, akkor azt közöljük. Olvasóink humán feladatok számítógépes megoldásának közlését is várják, nemcsak játékprogramokét; ilyen cikkeket is kérünk.

Nem kevés kritikát is kaptunk. Ennek örülünk, mert a segítő szándék nyilvánvaló.

Sokan szóvá tették, hogy miért közlünk hirdetéseket. Nem titok, a lap anyagi bázisának nagyobbik részét a hirdetések biztosítják. Ha nem lenne hirdetés, a lap árát nem tudnánk tartani. Arra buzdítjuk hirdetőinket, hogy ne reklám ízű szövegeket, hanem hardver- és szoftvertermékeik jó és érdekes leírását közöljék. Talán nem is eredmény nélkül.

Olvasóink egy része a tanfolyamot túl alapfokúnak, mások pedig nem eléggé alapozónak, inkább magas színvonalúnak tartották.

Megrovást kaptunk, hogy nagyon sok idegen szót, kifejezést használunk (még a nevünkben is!). Mit csináljunk, ha számítástechnikai nyelvünkben kevés a jó és találó magyar kifejezés. Tervezzük, hogy mozgalmat indítunk a szakmai nyelv magyarítása érdekében. Magyarítást, de nem minden áron – akkor inkább vállaljuk az elmaradtalást az idegen kifejezések miatt.

Néhányan azt szeretnék, ha a lap kevésbé mozgalmas, egy kissé konzervatívabb lenne. Mások, ha élénkebb formában, „magazinosabban" szerkesztenénk. Egyesek több, mások kevesebb képet és több „használható szöveget", például programleírásokat, kódtáblázatokat és más hasznos dolgokat kértek.

Tanárok kérték – többen is –, hogy az Iskola – számítógép rovatban közöljük más tanárok írásait, hogy hogyan alkalmazzák az iskolai számítógépeket különböző tantárgyak oktatására. Csak küldjék.

Olvasóink azt is szóvá tették, hogy kevés alkalmazási példát közlünk. Ne higgyük, hogy csak a játékprogramok érdeklik az olvasókat; arra is kíváncsiak, hogy a házi számítógépeket hogyan tudják használni a gazdaságban, kisvállalatoknál, decentralizált szervezetekben. Sokan a házi számítógépeken végzett gyakorlatokkal szeretnének előkészülni a professzionális számítógépen való komolyabb munkára. Ezért fogadják szívesen termékismertetőnket, ahol folyamatosan bemutatjuk elsősorban a hazai gyártású házi, illetve professzionális számítógépeket.

Olvasóink „harcra" is biztatnak bennünket: próbáljuk meg a ma érvényes „esztelen" személyiszámítógép-árakat letörni. Indítsunk mozgalmat, hogy változzék meg a behozatalt nehezítő vámrendelet, próbáljuk az illetékeseket meggyőzni, hogy a számítástechnika társadalmi elterjedésének egyetlen lehetősége, hogy sok gép jön be az országba. Megpróbáljuk.

Aztán egy sor új javaslatot is kaptunk, mint cikkek átvételét hasonló külföldi lapokból (CHIP, Byte stb.). Közöljük számítógépek tesztjeit, adjunk értékeléseket, nemzetközi összehasonlításokat, közöljünk összefoglalót az alkatrészpiacról, a beszerzési lehetőségekről. Hozzuk létre a μ M mellett a házi számítógépek központi programarchívumát, és tároljuk, másoljuk, forgalmazzuk ezeket a programokat. Annyi már történt, hogy a μ M-ban magánszemélyek és intézmények programjait, hirdetéseit közöljük. Lehet, hogy a következő lépés a programarchívum lesz?!

Azt hiszem – annak ellenére, hogy a levelekben küldött javaslatoknak csak egy részét tudtam leírni –, látszik: olvasóink törődnek a lappal, féltő szeretettel, sok ötlettel és javaslattal segítik szerkesztőségünk munkáját.

Végül meg kell köszönnöm, hogy olyan komoly sajtóorgánumok, mint a Heti Világgazdaság és a Magyar Nemzet, valamint a Magyar Rádió és nem utolsósorban a Magyar Televízió, a Kalendárium, az Iskolatévé és a Stúdió '84 adásai megértették céljainkat, felkarolták kezdeményezéseinket – egyszóval foglalkoztak velünk.

Szeretnénk olvasóink bizalmának megfelelni, és közkedvelt, népszerű, színvonalas lapot szerkeszteni, ahogyan ezt az indulásknor megfogalmaztuk:

együtt az olvasókkal.

KOVÁCS GYŐZŐ

Az első, 1983-as számunkban a személyi számítógép meghatározásával és főbb jellemzőivel foglalkoztunk. Most a személyi számítógépek főbb alkotóelemeivel ismerkedhetünk meg egy kicsit részletesebben.

Mi a személyi számítógép?

A mikroprocesszor

A mikroprocesszor a személyi számítógép központi egysége. Aritmetikai és logikai műveleteket hajt végre, továbbá regisztereket és az egész rendszer működését vezérlő elemeket tartalmaz. Egyetlen integrált áramköri tokban helyezkedik el.

A mikroprocesszor teljesítőképességét szóhossza és elektronikus órájának frekvenciája határozza meg. Szóhosszon a mikroprocesszor feldolgozási szélességét értik. Az első mikroprocesszor 4 bit szóhosszúságú volt (Intel 4004). Ma a 8 bites mikroprocesszorok a legelterjedtebbek (például az Intel 8080, a Z80, a Motorola 6800). Igen gyorsan terjednek azonban a 16 bites mikroprocesszorok (Intel 8086), sőt építenek már személyi számítógépeket 32 bites mikroprocesszorral is (Motorola 68000).

Az elektronikus óra a mikroprocesszor egyes műveleteinek időzítését vezérli. A szóhossz és az elektronikus óra frekvenciája – a megengedett maximális frekvencia, amellyel a mikroprocesszor még működni tud – által a mikroprocesszorok teljesítménye egyre nő. A szóhossz növekedésével egyre kevesebb gépi ciklussal lehet végrehajtani az egyes műveleteket, a frekvencia növekedésének eredményeként viszont nő az egységnyi idő alatt végrehajtható gépi ciklusok száma. A 16 bites mikroprocesszorokra épülő személyi számítógépek teljesítménye kb. egy nagyságrenddel nagyobb a 8 bitesekénél, és nagyjából azonos az eltérés a 32 bitesek teljesítménye és a 16 biteseké között is.

A nemrég bejelentett és az év második negyedévében megjelenő személyi számítógép, az IBM PC XT/370 már több funkcionális mikroprocesszort tartalmaz. Az együttműködő, a feladatokat egymás között megosztani képes mikroprocesszorokra épülő eszközök feldolgozóképessége olyan nagy lehet, hogy egy ilyen személyi számítógépet az asztalára helyező ember egy mai középkategóriájú számítógéppel azonos teljesítményű eszközt tudhat majd a magáénak. (Az IBM PC XT/370 teljesítménye a bejelentés szerint 0,4 MIPS – millió utasítás másodpercenként – lesz, ami az IBM 4331-es számítógép teljesítményének a fele.)

A központi tár

A személyi számítógépekben ma kizárólag félvezető tárolókat alkalmaznak. A tároló legkisebb egysége a cella: egy-egy cella 1 bit információt tárol. Maga a tároló ilyen mikroelektronikus cellák halmaza. Ezeket az elemi cellákat mátrix formában helyezik el. A tároló szervezési módját és kapacitását a mátrix dimenziói határozzák meg. A mátrix oszlopainak száma megadja, hogy hány bites egységenként lehet elérni a tárolóban elhelyezett információkat, a mátrix sorainak száma pedig a tárolóban elhelyezett ilyen egységek számával azonos. A sorok és oszlopok szorzata a tároló kapacitását adja bitekben. Tipikusak az 1, 2, 4 és 8 bites szervezési módok, azonban több tároló IC összekapcsolásával tetszőleges szervezésű memória alakítható ki. A félvezető elemek – chipek – tárolókapacitása a kezdeti 256 bitről 256 kbitre növekedett. Ma a legtöbb berendezésbe a 64 kbites elemeket építik be.

Egy tipikus professzionális személyi számítógép-konfiguráció

A tárolókat felülírhatóság szempontjából két csoportra osztják: az ún. véletlen hozzáférésű tárolókra (RAM) és a csak olvasható tárolókra (ROM). Az utóbbiakat – mivel tartalmuk programból nem változtatható meg, és a tápfeszültség kikapcsolása után is megőrzik a gyárilag beírt információt – elsősorban a személyi számítógépek vezérlő tárolóiként használják. Léteznek azonban olyan személyi számítógépek is, amelyekhez ROM-ba égetett – égetésnek nevezik a ROM-ba történő gyári beírást – alkalmazói programokat árulnak, amelyeket az erre a célra kialakított helyre beillesztve, a bennük elhelyezett program úgy használható, mintha azt egy háttértárolóról a RAM-ba olvastuk volna.

A RAM-ot – mivel tetszőleges elemeit ki lehet olvasni, és bármikor más információval felül lehet írni – írható-olvasható memóriának is nevezik. A háttértárakon elhelyezett programok és adatok beolvashatók a RAM-ba, a feldolgozás folyamán különböző részei módosíthatók. A RAM tárolók a bennük elhelyezett információt a tápfeszültség kikapcsolásakor elveszítik.

A személyi számítógépek tárolói a kezdeti 1–4 kbájtról már kb. 2 Mbájtra nőttek. Az egyszerű otthoni számítógépek (home computer) tipikus tárméretе ma 16–64 kbájt, a professzionális személyi számítógépeké néhány száz kbájt, a számítástechnikai célú személyi számítógépeké 1–2 Mbájt között van.

A háttértárolók

A személyi számítógépek háttértárolóiként eleinte szinte kizárólag kazettás magnetofonokat használtak, amelyek még ma is hozzákapcsolhatók a legtöbb berendezéshez. Tárolókapacitásuk kb. 100–150 kbájt. A programok és adatok a mágnesszalagon egyetlen lineáris sorban helyezkednek el. A szalag sebessége viszonylag kicsi, így egy konkrét információ megkeresése és beolvasása lassú.

A személyi számítógépek legelterjedtebb információhordozója a hajlékony mágneslemez (floppy). 8"-os, 5,25"-os vagy 3,5"-os hajlékony Mylar korong, amelyet egyik vagy mindkét oldalán mágneses anyaggal vonnak be. Ennek megfelelően beszélhetünk egy-, illetve kétoldalas lemezekről. Az információt a lemezeken koncentrikus körökön helyezik el, amelyeket sávoknak nevezünk. A sávok szektorokra – egyenlő nagyságú körcikkekre – oszlanak. A mágneslemezeknél az egy szektornyi információ az a legkisebb egység, amelyet címezni, illetve egyetlen művelettel írni-olvasni lehet. A szektorok száma a használt formátumtól függően 8 és 26 között lehet. Minden szektor 128–512 bájtnyi információt tárolhat. A hajlékony mágneslemezen tárolható információ mennyisége függ a szektorok számától, az egy

szektoron elhelyezett információ mennyiségétől és a lemezen elhelyezendő sávok számától. Ma a tipikusnak tekinthető hajlékony mágneslemezeken 125–500 kbájtnyi információ helyezhető el.

A személyi számítógépet alkalmazók körében egyre terjed a Winchester háttértárolók használata. Ezeknél a mágneses réteget merev alumíniumlemezre viszik fel. A Winchester tárolólemez nem cserélhető. Gyárilag egyszer s mindenkorra a meghajtóegységbe szerelik. A két előbbi tárolótípusnál lényegesen több – jelenleg 6–50 Mbájt közötti – információ tárolására alkalmas, a kívánt információ elérési ideje sokkal rövidebb, az írás-olvasás sebessége lényegesen nagyobb, ára viszont még meglehetősen magas.

Megjelentek már mini cserélhető merevlemezes tárolók is, 5,25" méretben, 6 Mbájt tárolókapacitással. Ezek ma még szintén igen drágák, így alkalmazásuk sem válhatott eddig általánossá.

A megjelenítő eszközök

A személyi számítógépekhez általában képernyős megjelenítőket alkalmaznak, amelyek lehetnek – színes vagy fekete-fehér – monitorok vagy normál televízió-készülékek. A megjelenítendő betűk, számok, jelek képét pontmátrix formájában – 5 × 7 vagy 7 × 9 pont karakterenként – ROM tárolóban helyezik el, és ezekből állítják össze a RAM tároló kijelölt részén a képernyőre kerülő szöveget. Egy képernyőre 12–26 sort, és soronként 40–80 karaktert – betűt, számot vagy jelet – lehet kiírni.

A személyi számítógépek igényesebb példányai alkalmasak grafikus információk kezelésére is. A grafikus ábrázolás finomságát a felbontóképességgel jellemzik. Ez azt mondja meg, hogy egy képet hány pontból lehet összeállítani. Személyi számítógépeknél általánosan elterjedt a 640 oszlop × 200 sor felbontás fekete-fehér képeknél és a 320 oszlop × 200 sor színes képek esetében. A grafikus képek – különösen a színesek – kezeléséhez természetesen lényegesen nagyobb memóriaterület szükséges, mint normál, nem grafikus módban.

A technológia gyors fejlődésével, az árak csökkenésével – különösen kisebb, hordozható eszközöknél – egyre terjednek a folyékonykristályos és a gázkisüléses rendszerű kijelző eszközök is.

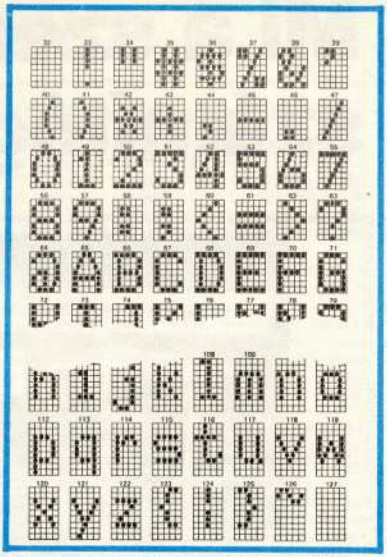
A billentyűzet

A felhasználó a billentyűzet segítségével adhat a személyi számítógépnek parancsokat, vihet be közvetlenül adatokat. Komolyabb személyi számítógépek az írógép billentyűzetéhez hasonló kialakítású vezérlő és funkcióbillentyűket tartalmaznak. Egyszerűbb és olcsóbb készülékek billentyűzetének kialakítása is kevésbé igényes, gyakran egyszerű fóliaérintkezős megoldású.

Nyomtatás

A személyi számítógépen végzett feldolgozás eredményére gyakran nyomtatott formában is szükségünk van. A nyomtatók igen széles választéka áll rendelkezésre (sajnos, ez a hazai viszonyokra nem jellemző). A nyomtatók a nyomtatott anyag minőségében, a nyomtatás sebességében és az árban különböznek egymástól.

A legolcsóbb nyomtatóeszközök a hőnyomtatók, amelyek a nyomtatandó képet speciális



Betűk, számok, jelek kialakítása pontokból a mátrixnyomtatókkal

papírba égetik kb. 50 betű/másodperc sebességgel. Gyorsabbak, drágábbak a mátrixnyomtatók. Ezeknél 7–18 tű halad végig a papír fölött, és festékszalagra ütve, pontokból alakítják ki a papíron a kívánt betűket. A nyomtatás minősége ezeknél attól függ, hogy egy betűt hány mátrixpontból alakítanak ki. Sebességük 80–200 betű/másodperc között változik. Kaphatók színes grafikus változataik is.

A fenti nyomtatókkal készült nyomtatás minősége gyakran nem elégíti ki az igényeket. „Levél minőségű" nyomtatáshoz az írógépekhez közel álló, drágább nyomtatók kaphatók. Ezek az ún. repülőkerekes – más néven margarétakerekes vagy tárcsás – nyomtatók, amelyeknél a betűk egy forgatható tárcsa karjain helyezkednek el. Miközben a repülőkerék a papír fölött halad, olyan helyzetbe fordul, hogy a kívánt betű kerüljön a nyomtatási pozíció fölé. Amikor ez megtörtént, egy kalapács a kart egy festékszalagra üti. A tárcsa cserélhető, ami biztosítja, hogy e nyomtatókkal többféle betűforma – akár a kézíráshoz hasonló is – íratható. Áruk a mátrixnyomtatókénál lényegesen magasabb, de nyomtatási minőségük révén a személyi számítógépek a legigényesebb nyomtatott dokumentumok létrehozására is használhatóvá váltak.

Csatlakozások

A legtöbb személyi számítógépnek vannak külső csatlakozási lehetőségei. Ezek teszik lehetővé, hogy bővítsük a memóriát, különböző szabványos interfészeken keresztül nagyszámítógépekhez csatlakozzunk, több személyi számítógépet összekössünk, speciális berendezéseket – például laboratóriumi mérőeszközöket – illesszünk a személyi számítógéphez stb.

A hardver azonban önmagában még nem alkalmas a felhasználó problémáinak megoldására, ezért a személyi számítógépeket szoftverrel kell ellátni. Legközelebb erről írok.

FINTA GYÖRGY

Véleményem szerint

Nagy figyelemmel és érdeklődéssel olvastam végig a Mikroszámítógép Magazin 1983-as számában megjelent: „Mi a személyi számítógép?" című cikket.

Örülök, hogy a professzionális és a személyi számítógépek közötti különbségek már szélesebb körű érdeklődésre tartanak igényt. Ahogy a cikkből is következik, természetesen a paraméterek közötti különbségeket, illetve a két géptípus paramétereit nem lehet abszolutizálni. A technológia állandó fejlődésével a géptípusokra jellemző paraméterek is folyamatosan változnak. De vigyáznunk kell, nehogy a magyar ipar által gyártott, kimondottan professzionális számítógépeket (például M08X, TZ–80) félrevezető kategóriába soroljuk csak azért, mert nem rendelkeznek 64 kbájtnál nagyobb operatív memóriával.

A cikk első két fejezetében leírtakkal nagyjából egyetértek; a harmadikban és a negyedikben szereplő definíciókkal azonban már vitába szállnék.

Nem hiszem, hogy professzionális számítógépeknek csak azokat lehetne tekinteni, amelyeknek „operatív táruk 64-től néhány száz kilobájtig terjed", továbbá „olcsó nyomtató kapcsolható hozzájuk". Véleményem szerint a személyi és a professzionális számítógépek közötti különbségek sok esetben nem annyira élesek, hogy olyan mértékű filozófiát lehetne belevinni a különbségekbe, mint ami a cikkben szerepel (például a vásárolható szoftvereszközök vonatkozásában).

A személyi és a professzionális számítógépek közötti különbségeket, illetve ezek ismérveit a következőképpen tudnám összefoglalni. A személyi számítógépeket elsősorban nem azzal a céllal adják el, hogy olyan személyek használják, akiknek napi nyolc órában van szükségük a gépre. A gyártók feltételezése szerint gépeiken nem főállású profi programozók vagy nagy mennyiségű adat bevitelét és mozgatását végző operátorok dolgoznak. Elsősorban azoknak a pénztárcájára apellálnak, akik csak hétvégeken, ritkán használják a számítógépet, tehát akik nem azt nézik, hogy display helyett villódzó tv van, nincs megfelelő sebességű adatmozgatás (lemez helyett kazettás magnetofon stb.), hanem hogy lényegesen olcsóbban vehetik meg a számítógépet.

A professzionális számítógépek, mint ahogy a nevükből is következik, a profik részére készültek. Feltételezik, hogy egy profi programozó nem fog napi nyolc órában fólia- vagy gumimembrános tasztatúrával dolgozni, és hogy nem hajlandó két-három órán keresztül a villogó tv-t nézni. Ugyancsak fárasztó, ha az adatok mozgatása vagy az adattömbök rendezése nem profi módon, hanem lemez helyett kazettás magnetofonnal, lassan, nehézkesen történik.

Tehát hogy egy számítógép professzionálisnak vagy személyinek tekinthető-e elsősorban, szerintem nem attól függ, hogy 8 vagy 16 bites processzorból épül-e fel, vagy az operatív memóriája kisebb, nagyobb-e 64 kbájtnál, vagy hogy rajta van-e a hálózat szoftvere.

SZÉLES GÁBOR
Műszertechnika GMK

Új kiegészítő szoftver a VT20 számítógépcsaládhoz:

DS–80 többmunkahelyes
ntrögzítő rendszer

ntrögzítés egyidejűleg több
minálon
nyű képernyőformátum
nerálása
ögzített adatállományból
enőrzött szekvenciális
DABAS-rekordok képzése
zítés közben

BASIC-hez
assembly-szubrutincsomag – MENÜ
automatikus képernyőkezelő
rendszer

képernyő létrehozása és eltárolása
meződefiniálási lehetőség
eltárolt képernyő megjelenítése és
ezen belül I/O műveletek végzése
BASIC és assembly-rutinok töltése
és indítása programból
(programláncolás)
– opcionális szubrutingyűjtemény
kiegészíti a BASIC-interpreter
lehetőségeit:
gyors adatbázis-kezelő rutinok
szekvenciális file-ok törölhetősége
diszkterület direkt elérése
stb.

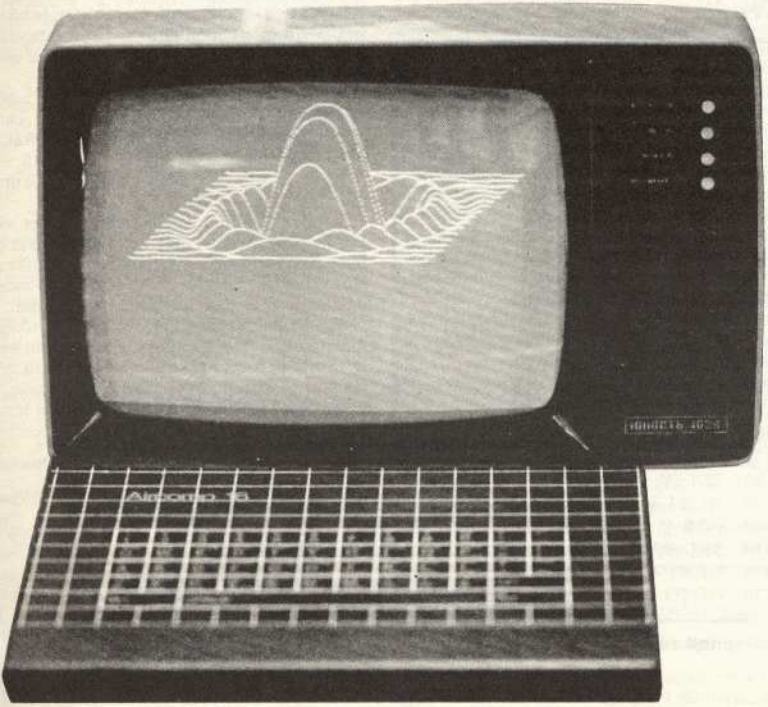
Mezőgazdasági programcsomagok

– Commodore VC–64, VT20 és VT20/A számítógépen:
Takarmányoptimalizálás
Tehénnaptár
Munkaügyi és bérelszámoló rendszer
Matematikai statisztikai programcsomag
Üzemanyag-elszámolási rendszer
– Fenti programok részben az AIRCOMP–16 és az AIRCOMP–32 számítógépen is futnak

Egyéb programtermékeink

Címlistakészítő több szempont szerint
pénztári címletező ellenőrzése
vállalati telefonkönyv készítő é
aktualizáló
grafikai kiértékelő rendszerek

Új, formatervezett kivitelben és megnövelt üzembiztonsággal szállítjuk az



AIRCOMP–16 személyi számító
olcsó
kiváló BASIC-interpreter
hatékony kiegészítő szoftver:
– dupla pontosságú aritmetika
– monitorbővítés assemblerrel
disassemblerrel és nyomköv
– játék- és oktatóprogramok

AIRCOMP–32 mikroszámítógépet:
programból történő magnóvezérlés
szükség esetén nyomógombos
billentyűzettel

URD–85 tv-alapú alfanumerikus
és grafikus terminált:
olcsó
24 × 80 karakteres képernyő
soros adatátviteli vonalon
összeköthető számítógéppel
képfelbontás 256 × 256 pont

LLALUNK kisszámítógépes rendszertervezést és komplett felhasználói programrendszerek írását.
Szaktanácsadás kisgépes rendszerek létrehozásában és a megfelelő géptípus kiválasztásában.

PERSONAL

AGROELEKTRONIKAI GT.
2040 Budaörs, Molnár P. u. 1.

Telefon: 260-612
Telex: 22-5962

Közös képviselő:
Agráripari Közös Vállalat,

2040 Budaörs, Nefelejcs u. 2.
Telefon: 260-612
Telex: 22-5962

BOSCOOP

Számítástechnika az oktatásban II.

BASIC nyelvű programok a HT–1080Z és a PC–1211 gépekre

Sorozatunk I. részében a középfokú oktatás szempontjából elemeztük a számítógépek jelentőségét és alkalmazási területeit. A fizikatanításra korlátozódtunk, és így is folytatjuk vizsgálatainkat. Néhány olyan kérdést tárgyalunk, amely a fizikatanítás központi részét képezi, és alkalmas arra, hogy a számítógép alkalmazásával továbbvigyük és ezáltal mélyebben vizsgálhassuk őket. Foglalkozunk olyan problémával is, amely kifejezetten a tanár-továbbképzést érinti.

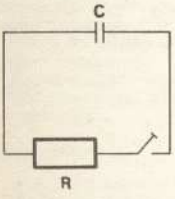
Kondenzátor aperiodikus kisülése

Az I. cikkben a japán PC–1211 (SHARP) típusú géppel tárgyaltuk a kérdést. Ezúttal a HT-1080Z iskola-számítógéppel tárgyaljuk, ami annyiban új, hogy a kisülés időbeli lefolyását grafikusan is ábrázolhatjuk, úgy, hogy berajzolhatjuk a koordinátatengelyeket, és be is számozhatjuk. Felhívjuk a figyelmet a szövegszerkesztés néhány jellegzetességére. Helyesebb READ, DATA utasítással, és nem INPUT-tal bevinni az adatokat, mert az előbbi esetben maguk az adatok a program részei, nem kell mindig újra beolvasni azokat. Különösen kényelmes ez az eljárás, ha sok az állandó adat, és csak egy vagy két adatot kívánunk újrafuttatáskor változtatni. Ezeket külön INPUT-tal vihetjük be.

A grafikus ábrázolást FOR TO NEXT utasítással valósítjuk meg úgy, hogy például csak minden N-edik lépést jelezzen ki, illetve vigyen fel a koordinátarendszerre a gép. A 110. lépéssel az ábrát jobb oldalról határoljuk le. A koordinátatengelyek kirajzoltatását és a vízszintes tengely beszámozását FOR TO NEXT és PRINT@ utasításokkal valósítjuk meg.

A 100. sorban a 2-es faktor, a 45 szám és a 0,4 faktor nem bír fizikai jelentéssel. Azért szükségesek, hogy a kezdő adatok fennállása mellett az egész képernyőt betöltő, vizuálisan jól megjelenített ábrát kapjunk: a kisülés negatív exponenciális feszültség-idő függvényét. Ezek tehát egy függvénytranszformáció alkalmasan választott konstansai.

A kezdőadatok és a programozható rekurziós képlet után felírjuk magát a programot.



$R = 1\ M\Omega.$
$C = 2\ \mu F.$
$t = 0{,}1\ s.$
$U_1 = 100\ V;$
$\frac{\Delta t}{RC} = 0{,}05;\ a.$
$\Delta U_k = -aU_{k-1}$
$U_n = U_1 + \sum_{k=1}^{n-1} \Delta U_k$

A 80. sor által visszük a programba a két, csatolt rekurziós formulát. A változók hozzárendelése:

$U \rightarrow A$

$B \rightarrow -\frac{\Delta t}{RC}.$

A pontosságot úgy növelhetjük, hogy Δt értékét csökkentjük, ezzel egyidejűleg nem jeleztetünk ki minden feszültségértéket, vagyis nem visszük fel a képernyőre. De tudjuk, hogy ezt nem is tehetnénk, hiszen a SET-tel megvalósított ábrázolásban csak egész értékenként jutunk új lépéshez X és Y irányban.

A 230. sorban végtelenített ciklust alkalmazunk, hogy ne jelenjék meg a képernyőn a READY üzenet, ami zavarja a kép jó láthatóságát. Ilyenkor BREAK-kal állítjuk le a futást.

A program:

```
10  PRINT "KOND. APERD. KISÜLÉSE"
20  READ A, B
30  DATA 100, – 0.05
40  INPUT N
50  FOR X = 0 TO 127
60  K = 0
70  K = K+1
80  A = A+A*B
90  IF K < N THEN 70
100 SET (2*X, 45–0.4*A)
110 IF X > 60 THEN 130
120 NEXT X
130 REM. INNEN A TENGELYEK
131 REM SZERKESZTÉSE
140 FOR U = 0 TO 60 STEP 5
150 PRINT@ (960+U), U;
160 NEXT U
170 FOR V = 0 TO 127
180 SET (V, 47)
190 NEXT V
200 FOR Y = 0 TO 47
210 SET (0,Y)
220 NEXT Y
230 GOTO 230
```

Csillapított rezgés vizsgálata

A harmonikus rezgés dinamikai alapegyenletéből indulunk ki:

$F = -DX$, ahol F a testre ható erő, D a direkciós erő és X a kitérés. Newton szerint ez esetben $F = ma$, vagy mx, így

$$a + \frac{D}{m}X = 0 \qquad (1)$$

a harmonikus rezgés dinamikai egyenlete. A csillapító erő általában a sebességgel arányos és vele ellentétes irányú: $F_{cs} = -kv$. Ezzel az

$$\ddot{x} + \frac{k}{m}\dot{x} + \frac{D}{m}X = 0 \qquad (1a)$$

differenciálegyenletre jutunk, amely a karakterisztikák módszerével egzaktul megoldható. Exponenciálisan lecsengő rezgést ír le az (1a) egyenlet. Érdekes, a látszólag egyszerűbb eset jelenti az igazi nehézséget, és a tanításban is ez fordul elő, csak nem megyünk mélyre a diszkussziónban. A látszólag egyszerűbb eset, mikor a csillapító erő – azaz a súrlódó erő – állandó nagyságú. Iránya továbbra is a sebességgel ellentétes.

Foglalkozzunk a továbbiakban ezzel, így elhagyjuk az (1a) differenciálegyenletet, és középiskolában még éppen járható gondolatmenetet követünk. Maga a probléma jó alkalom tanári elmélyedésre is.

Konstans csillapító erő esetén, amely tehát a tanításban is az elsőként tárgyalt $|F_s| = konst.$ ábrázoljuk az idő függvényében F_s értékét, ha ilyen erő csillapítja a rezgést.

Írjuk fel (1) módosított alakját, mikor F_s csillapít: $ma = -DX - F_s$;

F_s helyébe az ábra alapján annak Fourier sorát kell helyettesíteni. Az így kapott bonyolult differenciálegyenletet csak Laplace-transzformációval tudjuk megoldani, ha a közelítő megoldást analitikai alakban kívánjuk megadni.

Kínálkozik azonban egy igen jó lehetőség, a dinamikai egyenlet numerikus megoldása – gyakorlatilag nagy pontossággal. Ez az út ismét járható a középiskola szintjén. Fejezzük ki, hogy $|F_s|$ állandó, de mindig ellentétes irányú a pillanatnyi sebességgel. Ezt így tehetjük: $|F_s| \cdot \frac{v}{|v|}$. Ezek után vigyük ezt be a dinamikai egyenletbe, és a továbbiakban $|F_s|$ jelölésére egyszerűen F_s-et írunk. Ekkor

$$a + \frac{D}{m}X + \frac{F_s}{m} = 0. \qquad (2)$$

Bevezetünk egy közelítő módszert, amelynek alapján csatolt egyenletrendszert kapunk, két csatolt rekurziós formulát. A megtett út szempontjából linearizáljuk a mozgást, azaz elegendő kicsire választott Δt időközön belül egyenletes mozgással helyettesítjük a valódi mozgást. Az egyes Δt közökben érvényes sebességek számítását mindig az előzőben érvényes, egyenletesen gyorsuló mozgás feltételezésével végezzük. Így az intervallum-sebességek szempontjából a mozgást Δt közönként egyenletesen gyorsulónak tekintjük.

A Δt-ben az út szempontjából az egyenletes mozgás feltevése vagy a kondenzátorkisülés rekurziós formulái a matematikában Euler-módszer néven ismeretesek, mikor adott f(x) függvényt Δx közönként linearizálunk. Ezek után problémánkat így oldhatjuk meg:

$$\Delta x_1 = v_1 \Delta t$$
$$v_2 = v_1 - a_1 \cdot \Delta t$$
$$\Delta x_2 = v_2 \Delta t$$
$$v_3 = v_2 - a_2 \cdot \Delta t$$

Az *a* gyorsulás értékét (2)-ből helyettesítjük, és mindjárt az n-edik tagra írjuk fel a közelítést. A program alapjául szolgáló rekurziós formula végül ilyen:

$$\Delta x_{n-1} = v_{n-1} \Delta t \qquad (3)$$
$$v_n = v_{n-1} - \frac{D}{m} \Delta t \sum_{k=1}^{n-1} \Delta x_k - \frac{F_s}{m} \Delta t \frac{v_{n-1}}{|v_{n-1}|} .$$

A (3) formulában a második tagban

$$\sum_{k=1}^{n-1} \Delta x_k = X_{n-1} ,$$

azaz (n − 1) Δt idő alatt befutott (n − 1), egyenként különböző Δx_k szakaszok összege, tehát a t = (n − 1) Δt idő alatt létrejött kitérés. Éppen ezt jelenti (2)-ben az x mennyiség. A (3) rekurziós formula alapján készül a program.

Felsoroljuk a kezdő adatok választott numerikus értékeit és a kezdő adatoknak a BASIC változókhoz való hozzárendelését, majd következik a kilistázott program.

$v_1 = 12 \frac{m}{s}$	A = v
t = 0,03 s	B = Δt
D = 5 N/m	$C = \frac{D}{m} \Delta t$; 0,75
m = 0,2 kg	D = D
F_s = 1 N	$R = \frac{F_s \Delta t}{m}$; 0,15

A (3) formula alapján a képernyőn az x kitérés és a v sebesség, mint az idő függvényei jelennek meg. Jól látható az x(t) kitérés és a v(t) sebesség fáziskülönbsége.

```
10  PRINT "CSILL. REZGÉS"
20  READ A, B, C, D,
30  DATA 12, 0, Ø3, 0, 75, 5
40  INPUT R
50  G = Ø : H = Ø
60  FOR X = Ø TO 127
70  F = F + A * B
80  A = A – (C * F + R * A /ABS (A))
90  G = G + 1
100 H = H + 1
110 IF G < N THEN 170
120 G = Ø
130 IF H > 127 THEN 180
140 SET (Ø.6 * X, 10 * F + 24)
150 SET (Ø.6 * X, A + 24)
160 NEXT X
161 REM Ø, 6, 10 és 24 NEM FIZIKAI
162 REM JELENTÉSŰEK
170 GOTO 70
180 GOTO 180
```

Megjegyzés. A 70. sorban valósul meg a Δx_k értékek kiszámítása és összegezése, a 80. sorban az aktuális sebesség számítása. A 130. sor gondoskodik arról, hogy a kép ne nyúljon túl a képernyőn, míg a 180. sorral generált végtelen ciklus megakadályozza az END-del együttjáró READY üzenet kiírását – ha 180 END állna GOTO 180 helyett –, ami kissé belenyúlna az ábrába és zavaró lenne. Így csak BREAK paranccsal tudunk leállni.

A 40. sor szerkesztésével új futtatáskor mindig más-más R értéket vehetünk, azaz más-más F_s erőt anélkül, hogy ismét minden mennyiséget be kellene olvasnunk. Az INPUT kizárólag a megváltoztatni kívánt mennyiséget kérdezi. Így a READ, DATA, INPUT együttes kényelmes megoldást nyújt az R paramétertől függő görbesereg ábrázolásához.

Fontos, hogy R = 0 értékre az F_s = 0-ra beálló csillapítatlan rezgést kapjuk. Ekkor a 140. és 150. sorban új transzformációs konstansokat kell beiktatni – a régieket ezekre cserélni –, mivel olyan mértékűek a változások, hogy a grafikon túlnyúlik a képernyőn. Tehát a jelen megoldás lim x, lim v-vel az alapesetet is tartalmazza: a harmonikus R→0 rezgőmozgást.

Bessel-függvények előállítása PC–1211 géppel

Ez a probléma kifejezetten a tanártovábbképzés területére tartozik, és esetleg egy-egy kiemelt szakkörben tárgyalhatók fizikai vonatkozásai. A fizikában igen tág körben használhatók és nélkülözhetetlenek e függvények. A hővezetés különböző eseteiben, adott peremfeltételek mellett a probléma megoldását a Bessel-függvények adják. A folyadékáramlás területéről sok probléma kezelhető e függvényekkel, és még több az elektromágneses hullámok területéről. Az elektrosztatikában például töltött kondenzátor erőterének torzulása, inhomogenitása igen ötletesen tárgyalható a nulladrendű Bessel-függvényekkel.

Jó leírása megtalálható a Feynman-sorozat magyar fordításában, a 6. kötetben (141. oldal). Ez esetben arról van szó, hogy nagy frekvenciák hatására a lemezek közötti eredetileg homogén elektromos tér eltorzul az eltolási áram, pontosabban a jelen lévő E és H tér kölcsönhatása következtében. Az eredő elektromos tér E térerőssége a Jo(x) nulladrendű Bessel-függvény szerint alakul, ha x radiális távolságot jelent. Ebben a számításban a széleken fellépő torzulásokat nem vettük figyelembe. E néhány példa is arról győz meg, hogy igen fontosak e függvények. Rendszerint a vonatkozó probléma parciális differenciálegyenletének megoldó függvényeiként jelennek meg.

A p-ed rendű Bessel-függvényeket generáló sor az alábbi:

$$J_p(x) = \frac{\left(\frac{x}{2}\right)^p}{p!} + \sum_{n=1}^{\infty} \frac{(-1)^n \left(\frac{x}{2}\right)^{2n+p}}{n!\,(p+n)!}$$

A program úgy készült, hogy p≥0-ra érvényes, és p egész, továbbá könnyen tudjuk x és p értékét változtatni, így könnyen készíthető táblázat is.

A program felépítéséhez annyit, hogy a generáló sor szerkezete szerinti részekre bontjuk, és a faktoriálisokat szubrutinnal számítjuk. A szubrutinban két változót (X és Y) használunk. Nevezhetjük ezeket belső változóknak. A faktoriálisok kiszámítására ezekké alakítjuk át a p, n, (n + p) értékeket, és miután p!, n! és (n + p)! rendelkezésre áll, kiléptetjük X és Y-ból az n, p, (n + p) értékeket. (Jp(x)-ben x nem azonos az x belső változóval.)

```
10:  CLEAR
11:  INPUT "X = "; A
12:  INPUT "n = "; B
13:  INPUT "P = "; C
14:  M = B
20:  X = B : GOSUB 170
30:  D = Y
40:  X = B + C : GOSUB 170
50:  E = Y
60:  F = (A/2) ^ (2*B+C)
70:  G = B/2 – INT (B/2)
80:  IF G <> Ø LET F = –F
90:  I = F / (D*E) : H = H + I
100: B = B – 1
110: IF B > = 1 PAUSE B : GOTO 20
120: X = C : GOSUB 170
130: K = Y
140: L = ((A/2) ^ C) / K + H
150: PRINT "BESSEL = ";
     USING "# #, # # # #"; L;
     "␣ P = "; USING "# #"; C
151: PAUSE USING; L
155: H = Ø
156: INPUT R : IF R <> Ø
     LET A = R : B = M : GOTO 20
160: GOTO 11
170: Y = 1
175: IF X = Ø GOTO 210
180: Y = Y * X
190: X = X – 1
200: IF X GOTO 180
210: RETURN
220: END
```

Néhány megjegyzés:

- A program a 150. sorban kijelzi a keresett függvényértéket és azt is, hogy hányad rendű a Bessel-függvény.
- A 151. sorban egy pillanatra feloldjuk a rögzített pontos kijelzést.
- A 155. sor: H = Ø a független változónak, x-nek új értékkel való számításához szükséges, mert egyébként az új H-hoz hozzáadódnak H előbbi x-hez tartozó értékei.
- 14. sor: A 150-es leálláskor külön lehívással M, azaz M = B értékadás miatt n megnézhető, vagyis az, hogy a generáló sorból hány tagot vettünk figyelembe az éppen aktuális Jp(x) számításához.
- Ha új n és p mellett új x értékkel akarunk számolni, akkor a 156. sorban történő leállás után ENTER-rel indítunk, ami egyben a 156. sort érvényteleníti – átugorjuk –, így 160. által ismét a 11. sorra jutunk. Ez esetben a 156. sor INPUT kérdésére nem válaszolunk, hanem az ENTER-t billentyűzzük. Ha csak x értékét kívánjuk változtatni, akkor a 156. sorban INPUT után billentyűzzük az új x-et és ENTER-rel indítunk. Ekkor a 156. sor érvényes, és a 20. sortól kezdődik a számítás, miközben R lesz az új A, azaz x, és most itt kell még tárolni B = M értékadással n értékét.

DR. WIEDEMANN LÁSZLÓ

CSAPDÁK AZ RND KÖRÜL

A véletlen számokat előállító RND a HT–1080Z esetében a leggyakrabban használt utasítások egyike. Nélkülözhetetlen a statisztikus jelenségeket feldolgozó programokban, fontos eszköz a játékprogramok készítésénél. Gyakran kell bizonyos elemeket (számokat, betűket) véletlenszerűen sorrendbe rakni (permutáció). Problémamentes a feladat, ha az elemek ismétlődését megengedjük.

```
10 CLS:DEFINTI,K:INPUTK:FORI=1TOK:
  ?RND(K);:NEXT:END
```

Ez a programsor gyorsan és biztosan ad ismétléses permutációkat. K = 100-nál 2,4 másodperc alatt kapunk egy permutációt, de még a K = 10 000 is kivárható: 3 perc 40 másodperc.

Bonyolódik a dolog, ha az ismétlődést nem engedjük meg. Az ultizó bizonyára meglepődne, ha három piros hetest osztanának, és hibás lesz az a lottóprogram is, amely az ismétlődést nem zárja ki.

A megoldás kézenfekvőnek látszik: minden sorsolás után vizsgáljunk!

```
10 CLS:DEFINTA–Z:INPUTK:DIMA(K–1):
  A(Ø)=RND(K):?A(Ø);
20 FORI=1TOK–1:C=RND(K):FORJ=ØTOI–1:
  IFC=A(J)THEN40
30 J=I–1:NEXT:I=I–1:NEXT
40 NEXT:A(I)=C:?A(I);:NEXT:END
```

A program az első K db pozitív egész szám ismétlés nélküli permutációit állítja elő. És elérkeztünk az ünnepélyes pillanathoz: beleestünk az RND egyik csapdájába! Eleinte nem látszik a baj, a gép például K = 100-nál gyorsan kipotyogtat 50–60 db számot, de aztán egyre ritkábban sorsol olyat, amilyen még nem volt. Szegény embert még az ág is húzza! Ráadásul egyre több vizsgálatot kell elvégezni: a 80. számnál például 79-et. K elem ismétlés nélküli permutációját véletlenszerűen előállítva, a sorsolásszám várható értéke:

$$M = (k-1)!\left\{\frac{k}{k^{k-1}} + \frac{(k+1)(1+2+\ldots+k-1)}{k^k} + \right.$$

$$\left. + \frac{(k+2)[1\cdot 1+1\cdot 2+\ldots+(k-1)\cdot(k-1)]}{k^{k+1}} + \ldots \right\}$$

Összegzés helyett nézzünk néhány mérési adatot (HT–1080Z):

K	M	Idő	Kísérletek száma
10	28,5	2,16 mp	10
20	97,4	10,7 mp	10
50	224,6	49,3 mp	3
100	576,5	4 p 42 mp	2
200	1353	18 p 30 mp	2

K = 1000-nél e sorok írója csak becslésre vállalkozott: M kb. 10 000, a futási idő kb. 6–7 óra.

Az adatokból kitűnik, hogy a sok felesleges sorsolás és a nélkülözhetetlen vizsgálatok miatt a program – különösen „nagyobb" számok esetén – lassú. A javítás több irányban lehetséges.

Bármelyik permutációból bármelyikbe eljuthatunk úgy, hogy két elemet felcserélünk, és ezt a műveletet valahányszor megismételjük. Induljunk ki az 1,2,…,k permutációból, majd a számok cserélgetésével elérhetünk egy új sorrendet. Vigyázat, csapda a láthatáron! Hányszor cseréljünk?

5 szám permutációit láthatjuk az *1. ábrán*. Ezeket az 1,2,3,4,5 alaphelyzetből kiindulva, pontosan két cserével állítottuk elő. (Egy elemet önmagával is cserélhettünk.) Jól látható, hogy nem véletlenszerűek a sorrendek, például gyanúsan sokszor van az 1-es elől és az 5-ös hátul. Átlagosan kétszer illene egy számnak a helyén lenni. Bizonyos permutációkat két cserével el sem érhetünk. Az 5,4,2,3,1 például nem érhető el. Könnyen bizonyítható, hogy k elem esetén legfeljebb k–1 db cserével minden permutáció elérhető.

A program lehet tehát a következő:

```
10 CLS:DEFINTA–Z:INPUTK:DIMA(K–1):
   FORI=ØTOK–1:A(I)=I+1:NEXT
20FORI=1TOK–1:N=RND(K)–1
   :M=RND(K)–1:
   C=A(N):A(N)=A(M):A(M)=C:NEXT
30 FORI=ØTOK–1:?A(I);:NEXT:END
```

Valamit javult a helyzet. Az első módszernél 30-szor fordult elő, hogy egy szám a helyén szerepel, most csak 15-ször. (Ennek a számnak a várható értéke 10.) A fenti program még mindig nem állít elő teljesen véletlenszerű sorrendeket, hiszen azzal, hogy k–1-ben rögzítettük a cserék számát, a sok cserével előállítható permutációknak kedveztünk, a többiek rovására. Mennyi legyen akkor a cserék száma?

E kérdés megválaszolása nem célja jelen írásnak, helyette alkalmazzuk a kártyások örök igazságát: „Sokáig kell keverni, és a pakli jó lesz!"

Ha a fenti programban a 20 FORI=1TO2*K… módosítást alkalmazzuk, akkor a kapott permutációk jobban véletlenszerűek az eddigieknél. Most nyolcszor van egy szám a helyén. E program futási ideje:

K = 10 t = 1 mp
K = 100 t = 10 mp
K = 1000 t = 1 perc 48 mp

Az idő lényegében egyenesen arányos K-val.

Az egyik módszer korrekt, de nagyon lassú, a másik gyors, de a kapott sorrendek véletlenszerűsége megkérdőjelezhető. Található-e gyors és korrekt sorsolás? Található! Ne tegyük vissza azt az elemet, amelyet egyszer már kiválasztottunk. Ez így egyszerűen hangzik, de az RND visszateszi. Ne az elemet sorsoljuk az RND-vel, hanem a koordinátáját, az üresen maradt helyre pedig tegyük be a legutolsó elemet!

```
10 CLS:DEFINTA–Z:INPUTK:DIMA(K–1):
   FORI=ØTOK–1:A(I)=I+1:NEXT
20 FORI=1TOK:C=RND(K+1–I):?A(C);:
   A(C)=A(K–I):NEXT:END
```

A program 0,4 mp alatt készíti el 10 elem, 4 mp alatt 100 elem és kb. 40 mp alatt 1000 elem egy ismétlés nélküli permutációját.

A bridzselők kiválóan szórakoznak kiterített lapok kiértékelésével. A *2. ábrán* bemutatott program a fentiek alkalmazásával 18 mp alatt produkál egy „leosztást".

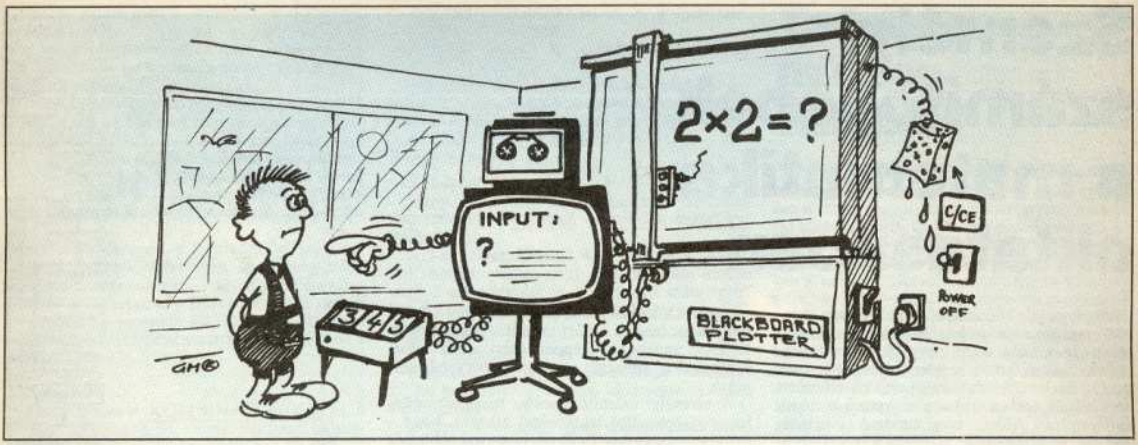
TOLVAJ LÁSZLÓ
tanár
Dr. Hetényi Géza Egészségügyi Szakközépiskola

1. ábra

2 1 3 4 5	4 2 1 5 3	1 5 4 3 2
2 1 3 4 5	4 2 5 3 1	1 4 5 2 3
1 2 5 4 3	5 3 1 2 4	5 4 1 2 3
4 2 1 3 5	2 1 3 4 5	3 1 2 5 4
4 2 3 1 5	4 1 5 2 3	5 1 3 2 4
1 2 5 4 3	1 5 2 4 3	3 5 2 1 4
1 2 3 4 5	1 4 2 5 3	2 3 4 5 1
1 4 3 2 5	1 2 3 4 5	1 2 4 3 5
4 3 2 1 5	1 5 4 3 2	3 2 1 5 4
1 2 3 4 5	5 2 4 1 3	2 4 1 3 5

2. ábra

```
10 CLS:DEFINTA–Z:DEFSNGG:DIMA(51):DIMN(12):DIME(12):DIMS(12):DIMW(12):DIMG(12)
20 FORI=ØTO51:A(I)=I:NEXT:FORX=49TO60:SET(X,24):NEXT:FORY=21TO27:SET(55,Y)
   :SET(54,Y):NEXT:?@667,"D";:?@544,"K";:?@534,"NY";:?@411,"E";
30 N=RND(52)–1:N(Ø)=A(N):A(N)=A(51):E=RND(51)–1:E(Ø)=A(E):A(E)=A(5Ø):
   S=RND(5Ø)–1:S(Ø)=A(S):A(S)=A(49):W=RND(49)–1:W(Ø)=A(W):A(W)=A(48)
40 FORI=1TO12:N=RND(52–4*I)–1:FORJ=ØTOI–1:IFA(N)>N(J)THENFORK=ITOJ+1STEP–1:
   N(K)=N(K–1):NEXT:N(J)=A(N):J=I–1:NEXT:A(N)=A(51–4*I):GOTO6Ø
50 NEXT:N(J)=A(N):A(N)=A(51–4*I)
60 E=RND(51–4*I)–1:FORJ=ØTOI–1:IFA(E)>E(J)THENFORK=ITOJ+1STEP–1:E(K)=E(K–1):
   NEXT:E(J)=A(E):J=I–1:NEXT:A(E)=A(5Ø–4*I):GOTO8Ø
70 NEXT:E(J)=A(E):A(E)=A(5Ø–4*I)
80 S=RND(5Ø–4*I)–1:FORJ=ØTOI–1:IFA(S)>S(J)THENFORK=ITOJ+1STEP–1:S(K)=S(K–1):
   NEXT:S(J)=A(S):J=I–1:NEXT:A(S)=A(49–4*I):GOTO1ØØ
90 NEXT:S(J)=A(S):A(S)=A(49–4*I)
100 W=RND(49–4*I)–1:FORJ=ØTOI–1:IFA(W)>W(J)THENFORK=ITOJ+1STEP–1:W(K)=
   W(K–1):NEXT:W(J)=A(W):J=I–1:NEXT:A(W)=A(48–4*I)GOTO12Ø
110 NEXT:W(J)=A(W):A(W)=A(48–4*I)
120 NEXT
130 P=24:FORI=ØTO12:G(I)=N(I):NEXT:GOSUB2ØØ:P=427:FORI=ØTO12:G(I)=E(I):
   NEXT:GOSUB2ØØ:P=792:FORI=ØTO12:G(I)=S(I):NEXT:GOSUB2ØØ:P=387:FORI=ØTO12:
   G(I)=W(I):NEXT:GOSUB2ØØ
140 GOTO14Ø
200 Z=Ø:L=Ø:X=Ø:Y=Ø:FORI=ØTO12:IFG(I)+1)/13=INT((G(I)+1)/13)THENC$="A":
   GOTO26Ø
210 IFG(I)+2)/13=INT((G(I)+2)/13)THENC$="K":GOTO26Ø
220 IFG(I)+3)/13=INT((G(I)+3)/13)THENC$="D":GOTO26Ø
230 IFG(I)+4)/13=INT((G(I)+4)/13)THENC$="B":GOTO26Ø
240 IFG(I)+5)/13=INT((G(I)+5)/13)THENC$="T":GOTO26Ø
250 C=2+G(I)–13*INT(G(I)/13):C$=STR$(C)/
260 IFG(I)>38THEN?@P+Z,C$;:Z=Z+2:NEXT:RETURN
270 V=P+64:IFG(I)>25THEN?@V+L,C$;:L=L+2:NEXT:RETURN
280 Q=P+128:IFG(I)>12THEN?@Q+X,C$;:X=X+2:NEXT:RETURN
290 R=P+192:?@R+Y,C$;:Y=Y+2:NEXT:RETURN
```



Oktatási programajánlat

Azonosító: TE/A01
Programnév: Szállítási feladat
Géptípus: ABC–80
Ára: 450,– Ft

A program a TECHNIKA II. tk. 130. oldalán lévő szállítási feladat programozott oktatását teszi lehetővé. Szemléletessé teszi az optimumszámítás lényegét. Jellegénél fogva a gimnázium matematikaoktatásához is felhasználható (első fokú egyenletek, függvényábrázolás). Tanári magyarázattal vagy tankönyv segítségével önállóan használható.

Azonosító: EG/H05
Programnév: Billentyű
Géptípus: HT–1080Z
Ára: 400,– Ft

Készségfejlesztő, gyakorlatot adó program. Segít az iskolai számítógép klaviatúrájának megismerésében, a BASIC-szavak és egyéb parancsok begyakorlásában, a gyors adatgépelésben.

Azonosító: FŐ/H02
Programnév: Magyarország
Géptípus: HT–1080Z
Ára: 450,– Ft

A program Magyarország megyéinek, megyeszékhelyeinek és 33 más – gazdaságilag jelentős – városának topográfiai begyakorlását segíti. Szakköri foglalkozásokra, valamint játékként történő felhasználásra alkalmas. Elvezhető formában gyakoroltatja a térképen való tájékozódást.

Azonosító: MN/A02
Programnév: GNII
Géptípus: ABC–80
Ára: 600,– Ft

A program a gimnáziumi magyar nyelv tantárgy oktatásának segítésére készült. A másodikos tananyaghoz kapcsolódva, a mondattan témakörben tesz fel ellenőrző kérdéseket. A pályázat 3 db programból áll. A programok váza azonos, csak a feldolgozott anyag más és más. Háromféle: feleletkiválasztó, asszociációs és kiegészítéses kérdéscsoporttal dolgoznak.

Azonosító: JÁ/A02
Programnév: Kertész
Géptípus: ABC–80
Ára: 600,– Ft

A program az osztott figyelem és a reflex fejlesztését elősegítő játék. A játékos feladata, hogy rovarölő spray segítségével megóvja a képernyő négy sarkában álló virágokat az őket állandóan fenyegető kártevőktől. Három virág elvesztése a játék végét jelenti, viszont minden elpusztított rovar (kukac és pók) után pont jár. A program 3 nehézségi fokozatban játszható.

Azonosító: JÁ/A03
Programnév: Hanoi-torony
Géptípus: ABC–80, HT–1080Z
Ára: 350,– Ft

A közismert Hanoi-torony játékot szimulálja a számítógépen. Egy vízszintes lapra három tartórúd van szerelve. A játék kezdetén az első rúdon 3–7 db, egyre csökkenő sugarú korong van. Feladat: a korongokat az első rúdról a harmadikra kell áttenni (a középső rúd felhasználásával) úgy, hogy egyszerre csak egy korongot lehet átrakni, és a leemelt korongot vagy üres rúdra, vagy egy nála nagyobb átmérőjű korongra lehet rátenni.

Azonosító: JÁ/H15
Programnév: Számháború
Géptípus: HT–1080Z
Ára: 400,– Ft

A sakkhoz hasonló logikát megkövetelő, koncentrálóképességet fejlesztő játék. Versenyrendszerben is felhasználható. Hagyományos táblás játékformában nem készíthető el.

Azonosító: JÁ/H17
Programnév: VAN GAM
Géptípus: HT–1080Z
Ára: 300,– Ft

A program izgalmas, szórakoztató logikai játék a számítógéppel. A „kupacos játékok" családjának legfiatalabb tagja. Van Withoff holland matematikus által alkotott algoritmus alapján készült.

Azonosító: JÁ/H21
Programnév: Színváltó
Géptípus: HT–1080Z
Ára: 300,– Ft

A program egy kevésbé ismert játék elvének ötletes felhasználásával készült. Gondolkodtató játék. A játékos többféle változat közül választhat. Egy 5 × 5-ös mezőt kell kifesteni egy mintával úgy, hogy közben a mezők állapota mindig változik. Cél, hogy minden mező világítson (a mezőnek két állapota van: világos vagy sötét). A minta középpontjának beadásával kezdődik a színezés. Nehéz a feladat megoldása, futása kb. 2–30 perc.

Azonosító: JÁ/H23
Programnév: Fekete doboz
Géptípus: HT–1080Z
Ára: 400,– Ft

Új, érdekes logikai játék. Jól fejleszti a logikai és kombinációs készséget. A kezdő állás kirajzolása után megkérdezi a belőtt golyó helyét. Kiírja egy táblázatba a belövés és kijövés helyét. Ha a lövés helyett new line-t nyomtatunk, akkor egy jelet a nyíl billentyűvel a feltételezett elrejtett golyó helyére írányithatunk. Ha ezen a helyen space-t nyomunk meg, akkor ott marad a jel. Négy golyónál több találgatást nem fogad el a gép. Minden játék után eredménylistát mutat. A futási idő 10 perc.

Azonosító: JÁ/H22
Programnév: Potty
Géptípus: HT–1080Z
Ára: 300,– Ft

Újszerű játékprogram. Jól fejleszti a logikai készséget, a bonyolult rendszerek áttekintését. Több személy játssza (kettő vagy három). Egyéni megértése nehéz. A kezdőállás kirajzolása után a potyogtatni kívánt betűt kell leütni, majd a billentyűkkel kell a nyilat mozgatni, azon potyogtatni. A futási idő 15–20 perc.

Azonosító: JÁ/H28
Programnév: NIM
Géptípus: HT–1080Z
Ára: 300,– Ft

A program feladata egyrészt, hogy bemutassa a NIM játékot és ezzel motiválja a NIM nyerő stratégiájának elkészítéséhez szükséges kettes számrendszert, továbbá a különböző logikai és Boole-műveletek tanítását; másrészt, hogy a játékon keresztül bemutassa a HT–1080Z számítógép lehetőségeit.

Azonosító: JÁ/H31
Programnév: Vadászok és nyuszi
Géptípus: HT–1080Z
Ára: 300,– Ft

Logikai készség fejlesztésére szolgál. Könnyen érthető a játék. Közép- és általános iskolába javasolt. A gép és egy játékos játssza, közismert sakktáblán. A vadászok mindig csak előre mehetnek, a nyuszik pedig bármilyen irányba a fekete négyzeteken. Cél, hogy a nyuszi kijöjjön a vadászok zárt láncából.

Azonosító: JÁ/H32
Programnév: FORDÍTS
Géptípus: HT–1080Z
Ára: 400,– Ft

A pályázat egy rendezési feladatra konstruált logikai játék. A program által felvetett probléma jó ösztönzést nyújt általánosabb kombinatorikai feladatok megoldására.

Azonosító: JÁ/A29
Programnév: Lépegető
Géptípus: ABC–80
Ára: 400,– Ft

A program a számítógéppel való ismerkedést segíti elő játékos formában. Előnye, hogy egyszerre két játékos játszhat vele. Minimális gépismeretet igényel, ezért kezdőknek is ajánlott. Szakköri alkalmazásra javasolt.

Személyi számítógépek a matematika oktatásában

Végre azzal büszkélkedhetünk, hogy Magyarországon minden középiskolának van legalább egy személyi számítógépe, ami nemzetközi összehasonlításban is jelentős eredménynek számít. Ezek a gépek azonban első közelítésben nem tudnak sokkal többet az egyszerű zsebkalkulátoroknál. Ahhoz, hogy tudásuk, lehetőségeik maximumát adják, még igen komoly munkára van szükség. Olyan programokat kell írni, amelyek segítségével szervesen bekapcsolódhatnak az iskolák vérkeringésébe.

Az iskolai életben a legnagyobb jelentőségük az oktatóprogroknak van és lesz a jövőben is. Szinte nincs is olyan tantárgy, amelynek tanításában ne lehetne valamilyen módon felhasználni, hasznosítani a számítógépet. Ezt az állítást számos külföldi és ma már néhány hazai tapasztalat is igazolja. Kétségtelen azonban, hogy új „büszkeségeink" legnagyobb szerepe, haszna a matematika oktatásában lesz. A számítógép a tananyag gyakorlatilag minden fejezeténél felhasználható mint bemutató, korrepetáló, ellenőrző tanári segédeszköz, vagy mint kísérletező, feladatmegoldást segítő eszköz.

A következőkben ezzel kapcsolatban szeretnék néhány gondolatot, ötletet elmondani.

A zsebszámológépek – és sokan a személyi számítógépek – alkalmazásától elsősorban azért félnek, mert szerintük a gyerekek már számolni sem fognak megtanulni, hiszen ezt elvégzi helyettük a gép. A probléma azonban nem ilyen egyszerű, hiszen a számolási készségnek, képességnek, a számolás szeretetének nagyon sok szintje és összetevője van.

Nézzük meg most ennek a kérdésnek csupán egy vonatkozását. A személyi számítógépnek azt az erényét, hogy igen jól tud számolni, azaz gyakorlatilag tévedhetetlen, nagyszerűen fel lehet használni többek között éppen arra, hogy számolni tanítson. A gyerekek tanulási gyorsasága egy osztályon belül is jelentősen különböző lehet, ezért a tanár képtelen a tanítási órákat úgy felépíteni, hogy azok minden diák számára egyformán érthetőek, követhetőek legyenek. Azoknak a tanulóknak, akik lemaradnak, akiknek nehezebben megy például az alapvető műveletek elsajátítása, nagy segítséget jelent egy olyan számítógépprogram, amely fokozódó nehézségű feladatokat ad a diáknak, rögtön ellenőrzi, hogy jól számolt-e, és azokat a részeket, amelyek nehezebben mennek, türelmesen (!) gyakoroltatja mindaddig, amíg szükséges.

A diák számára ennek sok előnye van. Az, hogy közvetlen kapcsolatot tud teremteni egy okos masinával, amely a tv-képernyőn feladatokat ad neki, és a klaviatúrán begépelt válaszokat gyorsan és tévedhetetlenül (!) ellenőrzi, kijavítja, lenyűgöző élmény a tanuló számára. A számítógép sohasem fárad el, nincsen „jó napja" és „rossz napja", és csak az érdekli, hogy a diák tudja-e a választ, vagy sem. Tehát nem személyes a kapcsolat az ember és a gép között. Ezért nem lehet a tanárt számítógéppel helyettesíteni, ugyanakkor éppen ezért jelent felbecsülhetetlen segítséget a számítógép a tanár számára.

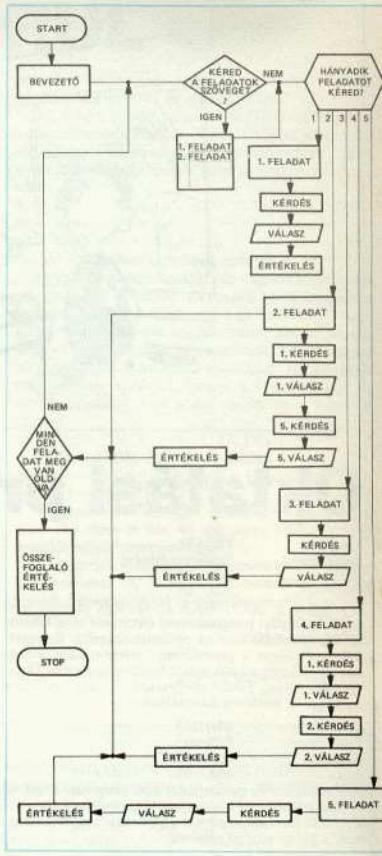
A személyi számítógépekbe beépített véletlenszám-generátor lehetőséget ad arra, hogy a programot úgy készítsük el, hogy azt többször lefuttatva, mindig más és más adatokkal adja meg a feladatokat. Tehát ugyanaz a tanuló ugyanazt a feladatsort, de különböző esetekben, annyiszor gyakorolhatja, ahányszor kell.

Ilyen gyakoroltató, felzárkóztatást segítő programot készíthetünk a matematika következő fejezeteinek számos részéhez is: halmazok, számelmélet, egyenletek, egyenletrendszerek, egyenletek közelítő megoldása, terület- és térfogatszámítás, trigonometria, kombinatorika, hatvány gyök logaritmus, koordinátageometria, sorozatok stb.

Amikor már nem a számolási készség fejlesztése a cél, e programok egy részét célszerű úgy elkészíteni, hogy futásuk megszakítható legyen, és a számítógépet mint számológépet is lehessen használni; a szükséges mellékszámítások gyors és pontos elvégzése után a program futása a megszakítás pontjától folytatható legyen. A HT–1080Z iskolaszámítógépen ez a BREAK billentyű és a CONT utasítás segítségével megtehető. Így egyrészt lerövidül a számolásigényes feladatok megoldásához szükséges idő, másrészt a tanulók megismerkednek a számítógép számábrázolásának alapjaival, bizonyos kerekítési problémákkal, és olyan típusú kérdésekkel, mint: mit jelent az, hogy egy végeredményt adott pontossággal kell megadni.

Már a legegyszerűbb programok esetén is óhatatlanul szembe kerülünk azzal a problémával, hogy a számítógép jelölései és a matematikában megszokott jelölések nem teljesen azonosak. Például a szorzás jele a matematikában a pont – a HT iskolaszámítógép erre a csillagot használja; a négyzetgyök jele helyett a gép a SQR jelölést, a tizedesvessző helyett pedig a pontot alkalmazza. A tapasztalat azonban azt mutatja, hogy ezeket a különbségeket a gyerekek gyorsan megszokják, és több-kevesebb gyakorlás után már nem jelentenek problémát. Sőt még előnyük is van, hiszen bizonyos ismereteknek a jelölésrendszertől független megértése általában magasabb szintű tudást jelent.

A tanítási órának is fontos szereplője lehet a számítógép, például a sok számolást igénylő, sorozatokkal, prímszámokkal, egyenletekkel, hatványozással, logaritmussal stb. kapcsolatos feladatok megoldásánál. A valószínűségszámításnál nagy számú kísérlet gyors elvégzésére, vagy a már említett véletlenszám-generátor segítségével jelenségek bemutatására is alkalmas. De lehetőség van arra is, hogy függvények grafikonjait kirajzoltassuk a képernyőre. Sajnos ezeknek a gépeknek nem túl jó a grafikus képfelbontásuk, de ennek is lehet haszna. Természetesen nem az alapoknál, hanem későbbi szinteken. Egyrészt tisztázódhatnak olyan kérdések (ha ez korábban még nem történt meg), hogy mit is jelent az, hogy megrajzoljuk vagy megadjuk egy függvény grafikonját, vagy ennek csak néhány pontját, és esetleg azokat is bizonyos hibával, másrészt pedig nagyon hasznos a tanulókkal olyan program készíttetése, amely adott függvények grafikonját rajzolja ki a képernyőre, de azt minél pontosabban, szemléletesebben. Egy ilyen feladat megoldása sokat segít a diákoknak a függvénytranszformációk lényegének megértésében is.



A számítógép új ellenőrzési lehetőséget is jelent a tanár és a diák számára egyaránt. Bemutatunk egy olyan *program-folyamatábrát*, amely dr. Korányi Erzsébet és dr. Urbán János Matematika IV. osztály című tankönyvének a 122–123. oldalán található ellenőrző feladatsor alapján készült. Ez a program 5 feladatot tartalmaz (ami szükség szerint növelhető szám). A program elindítása után a diák lekérheti a feladatok szövegeit a képernyőre. Miután kiválasztotta azt a feladatot, amellyel foglalkozni akar, a képernyőn megjelenik ennek a feladatnak a szövege. A kérdésekre adott válaszok után a gép azonnal értékeli is a megoldást.

Rossz válasz esetén – tetszés szerint (vagy éppen a feladattól függően) – történhet az, hogy a gép egyszerűen megadja a helyes választ, vagy az, hogy javasolja: próbálja meg a tanuló ismét megoldani ugyanezt a feladatot. Ezek után a diák kiválaszthatja a következő feladatot mindaddig, amíg az összeset meg nem oldotta. Ekkor a számítógép értékeli a diák munkáját, esetleg százalékosan a teljesítményét, vagy felhívja a figyelmét arra, hogy melyik feladatnál voltak hibák, és ezért a tankönyv melyik fejezetét, mely feladatait javasolja ismétlésre, gyakorlásra.

Most következzenek a feladatok leírásai, néhány megjegyzés kíséretében. A megjegyzések elsősorban arra utalnak, hogy a véletlenszám-generátor segítségével a feladatok egyes részeit – amelyeket a leírásban aláhúztunk – bizonyos korlátok mellett véletlenszerűen írathatjuk ki a képernyőre.

1. feladat

Ekvivalens-e a KOMBINATORIKA és a HALMAZELMÉLET szavak betűinek halmaza?

A két aláhúzott szó véletlenszerűen helyettesíthető például az EXPONENCIÁLIS (13, 10), IRRACIONÁLIS (12, 9), EGYBEVÁGÓSÁG (2, 8), SZÁMRENDSZEREK (14, 9), LOGARITMIKUS (12, 11), TRANSZFORMÁCIÓ (14, 13) szavakkal. A zárójelben levő számok közül az első a szó betűinek a számát, a második a szó különböző betűinek a számát jelenti. (A második szám értéke általában megváltozik, ha csak a számítógépen meglevő betűtípusokat engedjük meg!)

2. feladat

Legyenek az A halmaz elemei a 27 osztói, a B halmaz elemei a 12 osztói.

1. kérdés: Hány eleme van az A halmaznak?

2. kérdés: Hány eleme van az $\overline{A}$ halmaz hatványhalmazának?

3. kérdés: Hány eleme van A és B Descartes-szorzatának?

4. kérdés: Sorold fel az A és B halmazok közös részének elemeit!

5. kérdés: Sorold fel az A mínusz B halmaz elemeit!

A feladatban az A és B betűk véletlenszerűen felcserélhetők. A két aláhúzott számot például a következőképpen érdemes generálni:

X = RND(9) : A = RND (9)•X :

B = RND(9)•X

3. feladat

Hány olyan 3 jegyű szám van, amelynek minden számjegye páratlan és minden számjegye különböző?

A 3-as helyett állhat 2, 4, 5, 6 stb. is, a „páratlan" helyett páros, vagy például 6-nál kisebb vagy 2-nél nagyobb is, a „minden számjegye" helyett pedig a nem minden számjegye is.

4. feladat

Egy úszóversenyen 6 versenyző indul.

1. kérdés: Hányféle végső sorrend alakulhat ki, ha nem lesznek azonos időeredményt elért versenyzők?

2. kérdés: Az esetek hány százalékában kerül egy kiszemelt versenyző az első 3 közé?

Az „úszóverseny" helyett állhat autó-, kerékpár- vagy futóverseny is, a 6-os szám helyett 5–500-ig bármelyik egész szám, a 3-as helyett a fenti értéknél kisebb, 1-nél nagyobb egész szám.

5. feladat

6 vendég egy kerek asztalnál vacsorázik. Hányféle ülésrend lehetséges, ha két ülésrendet akkor tekintünk különbözőnek, ha legalább egy személy olyan, hogy valamelyik szomszédja különböző?

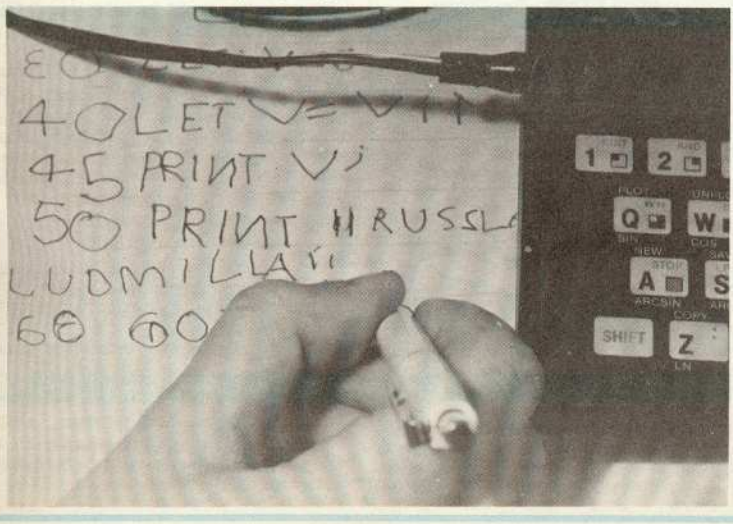
A vendégek száma elég nagy határok között mozoghat, és változhat a „legalább egy", illetve a „valamelyik" része is a feladatnak.

Végezetül álljon itt egy jól ismert mondás, kissé módosított formában:

Egy jó programot használni nagyon jó dolog,
de egy jó programot készíteni még jobb!

MAJOR ZOLTÁN
ELTE

Írni még nem tud, de a programja működik



Tanulja meg minden hallgató!

A számítógépnek mindennapi életünk számos területén fontos szerepe van. Néhány év múlva hazánkban is elmondhatjuk, hogy a számítógéppel kapcsolatos tájékozatlanság ugyanolyan problémákat vet fel, mint az írás és olvasás készségének hiánya.

A Kereskedelmi és Vendéglátóipari Főiskolán ebben a tanévben elkezdtük a személyi számítógépek (ABC–80 és HT–1080Z) kezelésének és programozásának oktatását. Első lépésként azt akarjuk elérni, hogy minden hallgató megtanulja a gép kezelését, használatát, és kellő gyakorlatot szerezzen kész, mágnesszalagon tárolt programok futtatásához. Az érdeklődőknek, elmélyedni kívánóknak lehetőséget adunk arra, hogy megtanulják különböző feladatok gépre programozását is.

Főiskolánknak jelenleg egy ABC–80-as gépe van, de hamarosan kapunk 10 darab HT–1080Z típusú iskolaszámítógépet is. Az oktatást már a gépek megérkezése előtt megkezdtük, a Fővárosi Pedagógiai Intézet és az ELTE Numerikus és Gépi Matematika tanszéke, pontosabban dr. Appel György és Kőhegyi János segítségével, akik aktívan közreműködtek a tematika kidolgozásában, és egyben az oktatást is vállalták. Lehetővé tették, hogy az FPI személyi számítógépes oktató kabinetjét használhassuk.

Az első fázisban az alábbi témakörök oktatását tervezzük, heti 3 órában:

1. foglalkozás: A személyi számítógép fő részei, helye a számítógép rendszerében. Gyakorlat: a gép üzembe helyezése, programbehívás.

2. foglalkozás: A BASIC elemei I. Az INPUT, LET, GOTO, PRINT utasítás. Elágazás: IF, THEN.

3. foglalkozás: A BASIC elemei II. Az értékadás más módjai. Stringek. Függvények.

4. foglalkozás: A BASIC elemei III. A grafikus ábrázolás lehetőségei.

5. foglalkozás: A grafikus ábrázolás gyakorlása.

6. foglalkozás: Véletlenszám-generátor. Játékfeladatok.

7. foglalkozás: A hanggenerátor működése.

8. foglalkozás: A számítógép alkalmazási lehetőségei a kereskedelemben, vendéglátásban. Típusprogramok.

9. foglalkozás: A számítógép alkalmazási lehetőségei a főiskolai matematika-gyakorlatokon.

10. foglalkozás: Összetett feladatok megoldása.

11. foglalkozás: Összetett feladatok megoldása, értékelése.

A későbbiekben az a célunk, hogy egyes statisztikai, vállalatgazdasági, szervezési problémák megoldására számítógépes programok készüljenek. A tanszék munkatársait és a jobb képességű hallgatókat arra ösztönözzük, hogy a kereskedelem, vendéglátóipar és idegenforgalom területéről számítógépes oktatási programokat készítsenek. A sikeres programokkal segíteni akarjuk a számítástechnika oktatását a kereskedelmi és vendéglátóipari szakközépiskolákban.

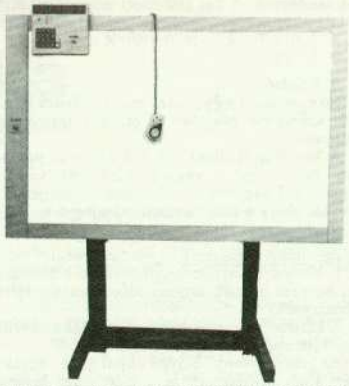
A főiskola oktatói körében nagy érdeklődést váltott ki az iskolaszámítógép. Alkalmazására számos lehetőség kínálkozik a képzésben, például árkalkulációk készítésénél, az idegenforgalmi földrajz oktatásánál, a készletelemzéseknél. De előbb persze meg kell ismerkedni a gépekkel és meg kell tanulni a programozást.

Reméljük, hogy terveink valóra váltásával – ha szerény keretek között is, – mi is hozzájárulhatunk a számítástechnikai kultúra terjesztéséhez.

LIGETI MÁRIA–CZÉTÉNY CSABA

Ha Önnek bármilyen grafikus ábrája van (fénykép, műszaki rajz, térkép, grafikon) és ezt számítógép segítségével akarja feldolgozni, akkor az Ön számára nélkülözhetetlen eszköz egy jó minőségű rajzdigitalizáló berendezés.
Csak néhány terület azok közül, ahol alkalmazása a számítógépes feldolgozás esetén nélkülözhetetlen: a tudományban a meteorológia, fizika, kémia, biológia, ahol a különböző mérések, vizsgálatok eredménye grafikusan, szalagdiagramon áll rendelkezésre. A kartográfiában, ahol térképek, térképvázlatok, légifelvételek által szolgáltatott információk tárolását és feldolgozását csak számítógéppel lehet elvégezni.
A gyógyászatban a különféle elektronikus érzékelők, diagnosztikai berendezések, röntgen fényképek által szolgáltatott információk feldolgozására.

Az iparban a különböző számjegyvezérlésű szerszámgépek és ellenőrző készülékek lyukszalagjainak előállítására, az elektronikai iparban a nyomtatott áramkörök készítésénél, a szabászatban, bőriparban a meglévő nyersanyag és a kivágandó felületek optimalizálásában, és még számos olyan területen, ahol grafikus ábrával kell számítástechnikai feldolgozásra alkalmassá tenni.
Mindezen feladatok elvégzésére alkalmas a *FOK-GYEM SZÖVETKEZET* által gyártott *RA–06 típ. INTELLIGENS RAJZDIGITALIZÁLÓ*



A RAJZDIGITALIZÁLÓ a következő részekből áll:

1. Munkaasztal: tartalmazza azt az X és Y irányú vezető elrendezést, amely érzékeli a pozicionáló eszköz által gerjesztett változó mágneses teret. A munkaasztal dőlésszöge változtatható.

2. Pozicionáló eszköz: egy tekercset tartalmaz, amely a vezérlőegységből jövő gerjesztőáram hatására változó mágneses teret hoz létre. A tekercs középpontjában található hajszálkereszttel kell a diagnosztizálandó rajzolatot lekövetni.

3. Display. Az egység tasztatúrájának segítségével lehet beprogramozni a digitalizáló működését meghatározó paramétereket, és ennek segítségével juttatható a digitalizáló adatbevitel, stop és mérés üzemmódba. A kijelzők az üzemmódoknak megfelelő információt mutatják.

4. Tasztatúra. Segítségével a DCII kódtáblázatnak megfelelő karakterek juttathatók a digitalizáló kimeneteire mérés üzemmódban, így a koordinátaadatokhoz tetszőleges kiegészítő információ rendelhető.

5. Vezérlő elektronika. A berendezés mikroprocesszoros vezérlésű. A mikrogép dolgozza fel az A/D által mért értékeket, vezérli a displayt és az interfész kártyákon keresztül a perifériákat, fogadja a tasztatúráról jövő karaktereket. Az elektronika egy párhuzamos kimeneti/bemeneti és egysoros aszinkron kimeneti/bemeneti interfésszel rendelkezik.

A digitalizáló üzemmódja:

1. Adatbevitel. Ebben az üzemmódban adhatók meg a működést meghatározó paraméterek:
– a koordinátaadatok mértékegysége mm vagy inch.
– felbontás X (mm): 0,1–2,5 0,1 mm-es lépésekben felbontás X (inch): 0,005–0,100 0,005 inch-es lépésekben
– felbontás Y (mm): 0,1–2,5 felbontás Y (inch): 0,005–0,100
– tengelyirány: a +X és +Y tengely hatféle módon vehető fel, ezáltal X és Y tengelyre való tükrözés és origó körüli 90, 180 és 270°-os elforgatás hajtható végre.
– adatkiviteli üzemmódban
Step: az X, Y koordináták a pozicionálón lévő Step nyomógomb megnyomásával juttathatók a perifériákra
Autó origin: a Step gomb megnyomásakor az előzőleg kivitt koordinátaértékekhez mért inkrementum kerül a perifériákra.
Contxy: a megadott kétjegyű szám első számjegye azt mutatja, hogy hányadik X, a második számjegye pedig azt, hogy hányadik Y raszter átlépésekor történik meg az X, Y koordináták perifériákra küldése.
Time: a beállított időintervallumtól függő időközönként kerülnek a koordináták a perifériára.
Remote: a párhuzamos vagy soros bemeneti perifériától függő időpontokban történik a koordinátakivitel
– Kimeneti blokkformátum: bináris
BCD
ASCII
– blokkhossz X: max. 8 digit és előjel
– blokkhossz Y: max. 8 digit és előjel
– nagyítás: max. 5 : 1
– kicsinyítés: max. 1 : 5
– kimeneti periféria: max. 2 párhuzamos (TTL szintű BSI) és 1 soros aszinkron (CCITT V. 24.)
A felsorolt paraméterek beállíthatók a display egység segítségével úgy, hogy minden paramétert külön megadunk, vagy a paraméterek egy adott összeállítását egy előre kitöltött táblázat egyik oszlopának meghívásával adjuk meg, de a beállítást elvégezheti a párhuzamos, vagy soros bemeneti periféria is.

2. Stop. Ebben az üzemmódban a pozicionáló szabadon mozgatható anélkül, hogy az utolsó mért koordinátaértékek megváltoznának. Ennek az üzemmódnak akkor van jelentősége, amikor a munkaasztal méreténél nagyobb rajzot digitalizálunk.

3. Mérés. A rajz pozicionálóval való végigkövetése ebben az üzemmódban végezhető el. A display a beállított paramétereknek megfelelő formában mutatja a pozicionáló origótól mért X és Y koordinátáját.

Műszaki adatok

Munkafelület (mm):	1050 × 675
Felbontóképesség (mm):	0,1 ± 0,02
Munkafelület méretének hőfokfüggése:	$2 \cdot 10^{-5}$/C
Letapogatási sebesség:	max. 1 m/sec
Számkijelzés X és Y irányban	±9999,9
Digitalizálható anyag:	max. 1 mm vastag nem mágneses anyag
Pozicionáló eszköz:	szabadon mozgatható tekercs, középen hajszálkereszttel.

További részletes felvilágosítás:
FOK-GYEM Szövetkezet,
Bp. XI., Karinthy F. u. 22.
Tel.: 666-703

Forgalmazza: MIGÉRT
Bp. VII., Népköztársaság útja 2.
Tel.: 323-322

LASSAN JÁRJ...

A „10 000 mikroszámítógép" program

A francia középiskolai oktatás nagyszabású, „10 000 mikroszámítógép" elnevezésű programja megvalósításának félidejénél tart.

A program létrejöttének előzményei 1970-re nyúlnak vissza. A francia kormány ekkor ismerte fel a középiskolai számítástechnikai oktatás fontosságát. Ekkor gondoltak először arra, hogy a számítástechnika hasznos segítséget nyújthat más tárgyak oktatásánál. Korábban csak az egyetemi számítástechnikai képzésre helyeztek súlyt.

Az emberi kultúra része

Felismerték, hogy a számítástechnika oktatását a középiskolákban kell elkezdeni, mert az az emberi kultúra része, és erre a tényre időben rá kell ébreszteni a diákokat. A cél tehát nem számítástechnikusok képzése, hanem a széles értelemben vett oktatás gazdagítása volt.

A francia oktatásügy 1970 és 1976 között „58 középiskola" néven kísérleti programot hajtott végre. Miniszámítógépet helyeztek üzembe 58 középiskolában, amelyeknél a kiválasztás feltétele a képzett felhasználók megléte volt. A munka során kifejlesztettek egy francia szimbolikus oktatási nyelvet, és 400 oktatóprogramból álló programkönyvtárat hoztak létre a Nemzeti Pedagógiai Kutatóintézet megbízásából.

A program végrehajtása során a legfőbb sikert a tanárok számítástechnikai oktatásban tapasztalt lelkesedése jelentette. A tanárok kétféle oktatási formában vehettek részt. Az első, amelyet magas szintű képzésnek neveztek, a program végéig 500 tanárt érintett. Az önként jelentkezők egy évre felmentést kaptak az oktatási kötelezettségek alól. Így nemcsak arra váltak alkalmassá, hogy munkatársaikat megtanítsák a számítástechnika oktatásban történő alkalmazására, hanem programokat is tudtak készíteni a hagyományos tantárgyak számítógépes oktatásához.

A másik tanfolyamtípus négyszer 2–3 napos oktatási periódusból állt. Ennek során a magas szintű képzésben részt vett tanárok adták át ismereteiket kollégáiknak. A program befejeztével 5 ezer tanár végezte el ezt a második tanfolyamtípust.

Az új technikára várva

A francia Oktatási Minisztérium 1976-ban leállította a programot, mivel a kísérletek során szerzett tapasztalatokat kielégítőnek találta. A program továbbfejlesztésének ügyével a Nemzeti Pedagógiai Kutatóintézet foglalkozott. Bár a továbbfejlesztést 1980-ra be akarták fejezni, a gyors technikai fejlődés miatt elhatározták, hogy a programot 1979-ig nem folytatják.

Az oktatás segítésében a számítástechnika alkalmazásának számos fontos és előnyös vonását ismerték fel. A legfontosabb az volt, hogy a számítógép használata elősegítette a tanulók logikus gondolkodását. A számítógép kiszélesítette továbbá a tanulók tudásterét, nemcsak a matematikai számításoknál, amelyek így sokkal hatékonyabban voltak taníthatók, hanem az általános gyakorlatban is, mert a diákok megtanulhatták a modellek használatát, amit a hagyományos osztálytermi oktatásban nem tudtak megtanulni.

A tanulók előtt végre megszűnt a számítógép titokzatos jellege, így a diákság megtanulhatta a gép előnyeit becsülni. Végül pedig lényegesen javult a tanár–diák viszony. Amint a tanítvány elveszítette a fonalat a gép használata során, kérhette tanára segítségét, aki így a lassú diákokra összpontosíthatott, miközben az osztály többi tagja folytathatta a párbeszédet a programozott tananyaggal. Ez az a pont, ahol a tanár bizonyíthatta, hogy a gép felett áll, amely a szövetségesévé vált; a tanulók tudták, hogy a program nem részrehajló. A tanár tehát olyan személlyé lett, akitől a tanulók kérdezhettek, aki velük együtt kereste a választ a gép által felvetett problémákra.

Itt aztán készültek a fogadásomra!

A francia kormány igen óvatos volt, amikor a számítástechnikát bevezettette a középiskolákban. Az „58 középiskola" program például a vidéki középiskolák alig 5 százalékát érintette, és ehhez is hatéves próbaidő állt rendelkezésre.

A hetvenes évek végén a számítástechnikában bekövetkezett gyors technikai fejlődés a berendezések árát lényegesen csökkentette, és ugyanakkor az alkalmazások rugalmasabbak lettek. Ez a kettős változás a francia kormányt már 1978-ban arra sarkallta, hogy további számítástechnikai oktatási terveit átértékelje, és új számítástechnikai programot indítson. A program finanszírozására az Oktatási Minisztériumot kérték fel 1979 februárjában.

Önálló tantárgyként

Az új program a korábbi időszak kedvező tapasztalatait figyelembe véve, alapjaiban kibővült: minden középiskolában mikroszámítógépek üzembe állítását tűzte ki célul.

A „10 000 mikroszámítógép" program első éve, 1979 végére már 400 új mikrogép működött az iskolákban. Ez a mennyiségi eredmény azonban nem az egyetlen új vonása a kibővült programnak. Amellett, hogy felülvizsgálták a kétféle, tanároknak szóló oktatási formát, a számítástechnika oktatása helyet kapott a középiskolai tanárok képzésében is.

1981 volt az első év, amikor a számítástechnika nemcsak az oktatás segítőjeként szerepelt, hanem arra is lehetőséget adott, hogy bizonyos iskolákban a diákok önálló tantárgyként tanulják.

A „10 000 mikroszámítógép" program költségei 1981–82 során elérték a 250 millió francia frankot.

A tervek szerint évente 200 újabb középiskola kap mikroszámítógépeket; mindenütt nyolc gépet és egy nyomtatót helyeznek el. A program befejezése 1987-re várható. Ekkorra mintegy 50 ezer tanárnak lesz majd számítástechnikai szaktudása, és a számítástechnika alkalmazása Franciaország valamennyi középiskolás diákja számára hozzáférhetővé válik.

KILIN JÓZSEF

Alapozás III.

Az információáramlás sajátságai

A számítógépbe információ kerül be, benne információtárolási, információáramoltatási és információfeldolgozási (transzformálási) műveletek zajlanak, és ezek eredményeképp információ kerül ki a számítógépből.

Természetesen az információbekerülés is és az információkikerülés is folyamat. Ezért helyesebb, ha úgy fogalmazunk, hogy a számítógépbe információ áramlik be, és a számítógépből információ áramlik ki. Az elsőt bemeneti információáramnak, a másodikat kimeneti információáramnak fogjuk nevezni. Minden áramlás eseményekkel (események valamilyen összességével) adható meg. Események viszont kijelentésekkel (ítéletekkel) írhatók le.

Célszerű, ha következetesen folyamatokkal igyekszünk dolgozni, folyamatokban igyekszünk gondolkozni, még olyankor is, amikor a folyamat csak egy-két különböző eseményből áll, mint például egy egyszerűbb program bemeneti információárama esetében. Vegyük szemügyre kissé részletesebben a számítógépben zajló legfontosabb folyamatokat, a leglényegesebb információáramlásokat!

Az információ tehát áramlik. E folyamatoknak, amelyeket áramoltatási és átalakítási műveletek hoznak létre, legfontosabb jellemzője az áramlási, illetve az átalakulási sebesség. A gyakorlat szempontjából különösen fontos szélső eset az információ 0 sebességű áramlása, más szóval az információtárolás. Az információfeldolgozás (átalakítás, transzformáció) vonatkozásában viszont a másik irányú szélső eset, a végtelen nagy sebességű (0 időigényű) művelet az érdekes. Természetesen csak elméletileg. Ezeknek, a valóságban soha meg nem valósítható műveleteknek azonban hasznuk van, mert alkalmas környezetbe helyezve, értékes elemei egy, a gyakorlatban jól alkalmazható elméletnek.

Az információ tehát ilyen vagy olyan sebességgel, így vagy úgy átalakulva áramlik, fluktuál a számítógépben.

Világos, hogy az információ e mozgásainak zajlása csak olyan helyeken történhet, amelyek alkalmasak arra, hogy ott az információ hosszabb vagy rövidebb ideig tartózkodhasson. Információtartózkodásra alkalmas helyek például a vezetékek, a mágnesszalag, a mágneslemez.

Nagyon fontos tudatosítanunk, hogy az információáramlást lehetővé tevő és megakadályozó helyek együttes, egymáshoz harmonikusan illesztett rendszere ad csak jól használható számítógépet, és önmagában se nem előnyös, se nem hátrányos ítélet például egy pontról az, hogy ott az információ szabadon áthaladhat, vagy lehetetlen az áthaladása. Egy nem kívánt érintkezés szabad információáramlást „biztosít"; itt az áramlás megakadályozása az érték. Egy nem kívánt szakadás esetében viszont fordított a helyzet. Jegyezzük meg tehát: az irányított, kézben tartott információáramoltatásnak nélkülözhetetlen feltétele, hogy legyenek információvezetést, áramlást lehetővé tevő (sőt lebonyolító) helyek is és információáramlást megakadályozó helyek is. Vegyük észre, hogy e „helyek" – nyilvánvalóan – műveletet végeznek, tehát lényegében operátorok.

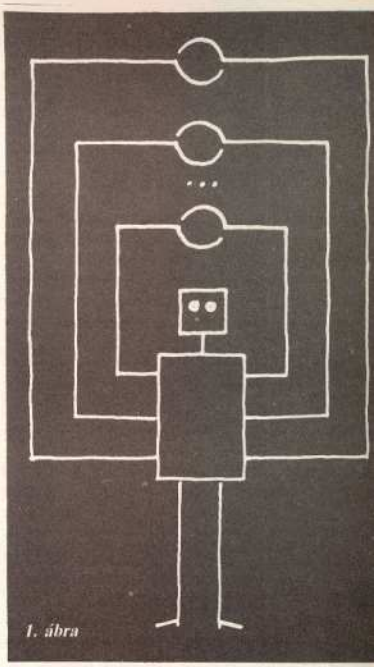
Ha egy információmozgási jelenséget le akarunk írni, azt tapasztaljuk, hogy ez semmiben sem különbözik más mozgási jelenségek leírásától. Minden esetben egymással kapcsolatban levő kijelentések rendszerével dolgozunk. Mindig helyre, időre stb. vonatkozó relációk, viszonyok (ezek is kijelentések!) viszik a főszerepet.

Az áramlások világa nagyon gazdag és érdekes világ. Sokféle áramlás van. Várható tehát, hogy legelőnyösebb tudományos megragadásuk módja sem lesz teljesen egyforma, és ez a mód igazodni fog az illető áramlás sajátságaihoz. Valóban így is van. Ezt a sokféleséget tapasztalhatjuk az információáramlások körén belül is. A számítógép esete több szempontból is jellegzetes. Egy ilyen sajátság a következő.

A gépben az áramlás lüktető jellegű. Az információ úgy áramlik, hogy rövid, többnyire szabályos időszakonként nekilendül, megmeglódul és meg-meg„pihen". Így a gépen belüli legtipikusabb folyamat e „pihenőhelyek" (tárolóhelyek) közötti (esetleg transzformált), irányított információáramoltatás.

A vezetékeket úgy tekinthetjük, hogy azok pontjaiban az információ egy pillanatig sem pihen meg, ott „átszalad", csak 0 ideig tárolódik, 0 ideig tartózkodik. A pihenőhelyeken (a tárolóelemekben) azonban rövidebb-hosszabb (de nem zérus!) ideig tartózkodik, „pihen" az információ.

A következőkben minden ilyen nem zérus időtartamú tartózkodást lehetővé tevő, tárolási műveletet végző helyet, elemet regiszternek nevezünk, függetlenül attól, hogy mekkora a tárolóképessége, azaz hogy mennyi információ feljegyzésére – idegen szóval regisztrálására – alkalmas, és e regisztrálás mennyi (nem zérus) ideig tart vagy tarthat.



1. ábra

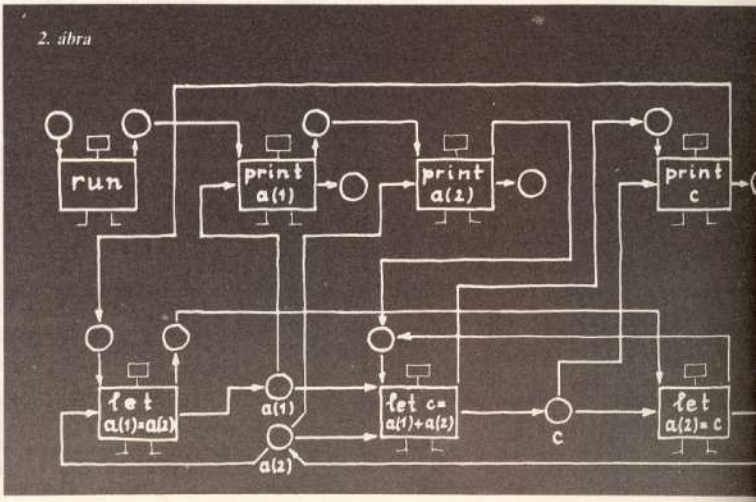
Kimondhatjuk tehát a mai (digitális) számítógép megértésének a kulcsát, ami a következő.

A digitális számítógépben regiszterek közötti (esetleg transzformált), tervszerűen irányított információáramoltatás folyik.

Így tehát a számítógép úgy is definiálható, hogy olyan regiszterek összessége, amelyek egy információáramoltatási és -átalakítási műveleteket végző rendszerhez kapcsolódnak.

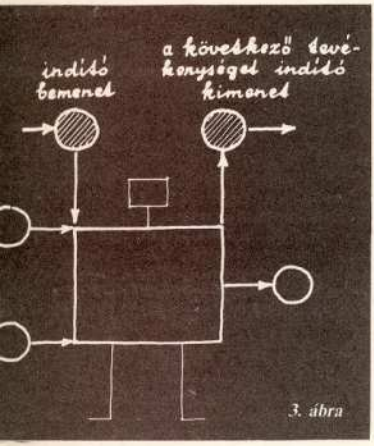
E rendszer a regiszterekből veszi az információt, és esetleg valamilyen átalakítást végezve

2. ábra



ezeken, ugyanezekbe a regiszterekbe helyezi el a műveletek eredményét.

A működés közben levő gép állapotát egy pillanatban csak a gép rövidebb-hosszabb múltbeli időszaka (működése és a bemeneti információáram) határozza meg. Ezt egy, az *1. ábrán* látható robottal szemléltethetjük. (A regisztereket, amelyeknek hatása van a működésre, a robot kezei jelzik. A robot átveszi a bal kezéből az információt a jobba, és elvégzi az előírt műveleteket, az eredményt pedig megjeleníti a bal kezében, majd (egy pillanat) szünetet tart. Esetünkben egy-egy információáramoltatási időszak végén a pihenő időszakbeli pillanatkép meghatározza a további működést (természetesen a bemeneti információárammal együtt). De ha a bemeneti információáram mindig csak regiszterek feltöltése formájában juthat be a gépbe – ahogy ez a legtöbb számítógép esetében ma így is van – és a számoláshoz szükséges információ bejuttatását (azaz a regiszterek „kezdeti" feltöltését) az érdemi számolás előtt teljes egészében elvégezzük, ami a technológiai folyamatirányító rendszereket és ezekkel rokon párbeszédes üzemmódban dolgozó rendszereket kivéve elvileg mindig megvalósítható,



3. ábra

akkor a regiszterek összességének állapotát egy „pihenő" időpontban, az egyvégtében elvégzett információ „behordást" követő időszakban, csak a regiszterek előző „pihenő" időpontokbeli állapota (állapotai) határozza (határozzák) meg.

Ilyenkor tehát a regiszterösszesség értéke egy pihenőpontban, előző értékekre támaszkodó, ún. rekurzív módon van definiálva, ugyanazoknak a regisztereknek az előző pihenőidőszakokbeli értékei által.

Folyamatdefiniálás saját és más folyamatok előző értékei segítségével

Vannak olyan folyamatok, amelyeknek bármely t időpontbeli értékét más folyamatok t időpontbeli értéke egyértelműen meghatározza.

Vannak azonban olyan folyamatok is, amelyek t időpontbeli értékének meghatározásához más folyamatok t időpontbeli értékén kívül azok múltja, azaz előbbi időpontbeli értékei is szükségesek.

A legáltalánosabb eset az, amikor egy folyamat t időpontbeli értékét e folyamat és más folyamatok t időpontbeli és múltbeli (előző) értékei egyaránt befolyásolják. Érdekes, hogy folyamatdefiniálás lehetséges jövőbeli értékekre vonatkozó információk felhasználásával is. Ezzel a kérdéssel azonban itt nem foglalkozhatunk.

Minden folyamatdefiniálási típusra adunk néhány jellegzetes példát.

Az első osztály legtipikusabb eleme a két folyamat összegeként előálló harmadik folyamat.

A második osztály, noha rendkívül népes, és sok fontos elemet tartalmaz, valamilyen ismeretlen okból teljesen mellőzve van, pedig – nézetünk szerint – a matematika megújulásának, további fejlődésének valóságos kincsesbányájaként, különleges becsben kellene állnia. Csupán ízelítőül néhány ide tartozó operátor, illetve definíció. Az a(t) folyamat t időpontbeli értéke lehet például a b(t) folyamatnak valamilyen meghatározott időponttól a jelenlegi t időpontig terjedő időszakra vonatkozó valamilyen értékeloszlás-jellemzője (maximuma, minimuma, mediánja, átlaga, entrópiája stb.), vagy ívhossza, vagy teljes megváltozása, integrálja stb.

A harmadik osztályba tartozik a következő definíció:

$$a(t) = \int_{1}^{t} b(x)a(x-1)dx, \qquad t \geq 1$$

(a(t) értékét a [0;1] szakaszon előre meg kell adni.)

E nélkülözhetetlenül fontos fogalmak jobb megértése érdekében további olyan difiníciós példákkal is foglalkoznunk kell, amelyek egyszerűség kedvéért most csak sorozatokra (vagy ha tetszik, lépcsős függvényekre, illetve csak egész időpontokban értelmezett függvényekre) vonatkoznak.

Egyszerű példa sorozatdefiniálásra

Legyen az *a* folyamat (sorozat) értéke egy (egész) időpontban egyenlő az *a* folyamatnak ezt az időpontot közvetlenül megelőző két (egész) időpontbeli értékének összegével. Formulával: legyen

a(n) = a(n−1)+a(n−2), (n = 3, 4, ...).

Természetesen a(1) és a(2) értékét külön meg kell adni.

Ennek az ún. Fibonacci-sorozatnak a kiszámítását – ha a(1)-nek és a(2)-nek már van értéke – a következő operátorcsapattal *(2. ábra)* és BASIC programmal végezhetjük el.

```
1Ø print a(1)
2Ø print a(2)
3Ø let c = a(1)+a(2)
4Ø print c
5Ø let a(1) = a(2)
6Ø let a(2) = c
7Ø goto 3Ø
8Ø end.
```

Mind a program, mind pedig az ábra szokatlan dolgokat tartalmaz. Először is mindkettő „örökmozgó". E program vezénylete alatt se a számítógép, se az operátorcsapat elvileg sohasem áll meg.

Ha valaki játszani kívánna e sorozattal és futtatni akarná a programot, akkor célszerű ezt két input utasítással kiegészíteni:

```
5 input a(1)
6 input a(2)
```

Az operátorcsapatban van egy run, ami a programban nincs. Ennek oka az, hogy szinte mindig a legkisebb sorszámú utasításnál kezdjük a program végrehajtását, amit mindig a run bebillentyűzése indít, ezért nem szerepeltetjük a programban a runt, ami a BASIC szabályok szerint tilos is. (Van azonban egy másik, nem haszontalan indítási lehetőség is. Ha például a 12Ø-as sortól kívánjuk a programot indítani, ezt a run 12Ø bebillentyűzésével érhetjük el.)

Az ábrán szereplő print operátorok kimeneti „kezében" mindig a képernyő utolsó sorának tartalma van. (Egyszerűség kedvéért a teljes nyomtatási mechanizmus ábrázolásától eltekintettünk.)

Valójában e print operátorok nem teljes értékű nyomtató operátorok. A nyomtatáshoz ugyanis meg kell adni, hogy mi és hova kerüljön rögzítésre. E print operátorok lényegében csak a nyomtatandót határozzák meg, azaz nyomtatnivalóval látják el az érdemi nyomtatást végző mechanizmust, amely automatikusan határozza meg a nyomtatandók és más kinyomtatottak pillanatnyi helyét, és végzi el a nyomtatást.

További furcsaság, hogy a programban van egy „goto", amelynek megfelelőjét viszont hiába keressük az ábrán. A rejtély megfejtése nagyon egyszerű: mind a programban, mind az ábrán érdemben jelen van a „goto", mégpedig nem is egy példányban. Láthatóan azonban csak egy helyen kellett megjelenítenünk, a program 7Ø jelű sorában, amely lényegében a 6Ø jelű sorhoz tartozik. Mint már erről szó volt, minden esetben egyértelműen meg kell mondanunk, hogy egy művelet befejezése után mit kell végeznünk. Más szóval: minden műveletben mindig érdemi szerephez jut egy „goto", mégpedig nem külön, hanem a művelettől elválaszthatatlanul. Ha a következő elvégeztetni kívánt művelet az éppen elvégzettet nagyságban közvetlenül követő számú, akkor e „goto" kiírását csupán kényelmi és helytakarékossági okokból mellőzzük.

Keressük most meg a „goto"-kat az ábrán! A „goto" olyan operátor, amely figyeli és jelzi egy művelet befejeződését, és e jelzéssel indítja (indíthatja) egy következő művelet végrehajtását. E feladat ellátását az ábra operátoraira bíztuk. Az ábra operátoraiba „beépítettük" a „goto"-t, úgy, hogy csak az operátorok jobb felső sarkából kandikál ki a beépített „goto" indító kimenete. Ezek az indító kimenetek kapcsolódnak az indítandó operátorok indító bemenetére, amelyeket az operátorok bal felső sarkain találhatunk *(3. ábra)*.

Egy operátor, ha indító bemenetén indulást kívánó eseményt tapasztal, hozzálát feladata elvégzéséhez. Ha feladatát befejezte, akkor „kész" jelet ad, azaz olyan jelenséget hoz létre, amelyet más operátor indító kimenetére vezetve, azt feladata végrehajtása megkezdésére kényszeríti.

Igen ám, de mi van olyankor, ha egy operátor új meg új indítást kap, de a feladatát (például összeadást) még nem fejezte be? És mikor mit értünk indító jelen? És mikor tekintünk egy operátort működésben levőnek?

Csupa olyan megválaszolásra váró kérdés, amely a lényeget érinti. Ezekkel foglalkozunk majd a továbbiakban.

POGÁNY CSABA

Mit tud

[az APL programnyelv]?

A külső környezettel (értsd: az operációs rendszer további részeivel; más paralel futó processzorokkal, perifériákkal) való kapcsolattartás ún. rendszerváltozókon és rendszerfüggvényeken (vagyis függvényeljárásokon) keresztül valósul meg. A külső környezettel függnek össze a segédprogramok is.

A rendszerváltozók kifejezésekben ugyanúgy szerepelhetnek, mint más változók; egy részük új értéket is képes felvenni. Lekérdezhető például a számlázási információ, a karakterkészlet vektora, a végrehajtási sorszámláló, az időpont-dátum, a termináltípus, a bejelentkezett felhasználók száma, a szabad munkaterület. Beállítható a nyomtatási sebesség, a tabulátorvektor, a nyomtatási pontosság alapértelmezése, a numerikus felbontás, a tömbök indexelésének alsó határa, a véletlenszám-generátor magja, valamint az ún. látens kifejezés, amely a munkaterület lehívásakor automatikusan végrehajtódik.

Az általános rendszerfüggvények a programokban ugyanúgy jelenhetnek meg, mint bármely más függvény. Ily módon dinamikusan, programvezérlés alatt kérdezhetők le objektumok jellemzői és törölhetők maguk az objektumok, nyerhető ki (és szerkeszthető) egy-egy eljárásdefiníció szövege, illetve a megfelelő szövegmátrix dinamikus eljárásként definiálható.

Specializált rendszerfüggvények szolgálnak külső processzorokkal való együttműködésre (shared variable processor). A kapcsolattartás a megszokott változók elvén nyugszik, és „kézfogásos" (handshaking) szinkronizálási módszert alkalmaz. Maga az elv, de a konkrét megoldása is teljesen általános: az APL rendszer és a külső processzor szimmetrikus szerepet játszik, és minden akcióhoz mindkettő aktív hozzájárulása szükséges. Így már a kapcsolatfelvétel is csak akkor jön létre, ha mindketten keresik a kapcsolatot, és erre a célra megfelelő változókat ajánlanak fel.

Az APL világszemlélete szerint nem szabad a nyelvbe nem standard eszközöket beültetni, bár ezek nélkülözhetetlenek (mint például fájlkezelés vagy grafika). Ezek megvalósítására kitűnő megoldás az önálló segédprocesszor, amely megosztott változókkal van csak kapcsolatban az APL nyelvi rendszerrel, egyébként teljesen független. Ezek az általános használatra szánt processzorok természetesen állandóan és mindenkinek felajánlkoznak (ebből a szempontból igencsak ledérek!), de lehetővé tesznek konkurrens használatot is. Ezzel a módszerrel realizáltak már fájlkezelőt, adatbázis-interfészt, mátrixprocesszor-csatolót, grafikus alrendszert, specializált perifériameghajtót, kommandeljárás-végrehajtót, képernyőformázót stb.

A segédprogramok a könyvtárak, munkaterületek karbantartási (kimentés-archiválás, visszatöltés, konverzió, rendszergenerálás) végzik. Segítségükkel a könyvtárak olyan standard formátumban archiválhatók, amely tetszőleges típusú gép APL rendszere számára érthető és visszatölthető.

Amit hiába keresel

Az APL nyelv első pillantásra legfeltűnőbb vonásai azok az eszközök, amelyek a megszokott nyelvi eszközökön felül jelentkeznek: a szokatlan szimbólumok, néhány páratlan operátor (például kódolás-dekódolás, mátrixosztás, de részben az „execute" és a formázás is), az összetett operátorok, a rendszerváltozók és rendszerfüggvények stb. További tanulmányozás után legalább ennyire feltűnőek azok a megszokott nyelvi eszközök, amelyek viszont hiányoznak.

Az APL nem ismeri a deklarációt, és ezzel együtt az explicite megadott típust sem. Helyesebb is azt mondani, hogy a változóknak nincs típusuk, csak belső ábrázolási formájuk, amely dinamikusan változik az értékadásokkor. Az ábrázolási forma összetevőiből az alak (a tömbök rendje, az egyes rendek mentén aktuális dimenzió) a programozó vezérlése alatt áll, a belső kódolás (egész, lebegőpontos szám, karakter, bit) viszont automatikusan konvertálódik a műveletek hatására, és a programozónak erről nem kell tudnia. (Már önmagában az a tény, hogy a változók alakja dinamikusan változik, rendkívül nehéz vállalkozássá teszi a compilerkészítést. Így nem csoda, hogy az APL nyelvi processzorok túlnyomó többségét értelmezőként valósították meg.)

Az APL nem ismeri a blokk-struktúrát; az egymásba skatulyázás egyetlen üzemi formájaként az eljárás-(függvény-)hívást ismeri el.

Az APL-ből hiányoznak a FOR ... STEP ..., a DO ... WHILE ... és hasonló konstrukciók. Egyáltalán: a strukturált programozás fegyvertára. Ezt súlyos hibaként szokták az APL szemére vetni. Készült is egy strukturált programozásra alkalmas változata (APLGOL), de nem terjedt el. (Személyes véleményem szerint a Dijkstra-féle strukturált programozás zsákutca, mivel az ember felfogó-áttekintő képességéhez Chomsky vizsgálatai szerint rosszul illeszkedik a többszörösen egymásba mélyített struktúra.)

Mint már utaltunk rá, a nyelvben nincs beépített fájlkezelő rendszer; a fájlkezelés segédprocesszorral oldható meg. (Megjegyzem mind a mai napig hiányoznak a megalapozott érvek a fájlkezelés standardizálhatóságára.)

Mikrók és az APL

Az első mikrogép, amit életemben láttam, az MCM70 volt, az 1975-ös pisai APL-kongreszszuson. Az APL történetének egészen kezdeti szakaszától kezdve élő téma az „APL-gép"; az MCM70 ennek első kereskedelmi realizálása volt. És nem is rossz, még mai szemmel sem! Kényelmesen befér egy aktatáskába, ugyanakkor az IBM teljes APL-reprezentációja, az APLSV teljes funkciókészletét tartalmazta. (Érdekes: az APL történetében soha senki nem próbált nyelvi részhalmazt fabrikálni!)

Azóta számos közkedvelt, elterjedt mikrogépen vált elérhetővé az APL (mint például a Commodore PET, CBM), de a komoly előrelépéseket mégsem a más nyelvi rendszerekkel osztozkodó APL-ek jelentették. Az igazi APL-kedvelőket nem nagyon érdekli a többi nyelv. Az APL külön világ; és ez a világ meglehetősen teljes, nem hagy hiányérzetet.

Nagy érdeklődést keltett, és hamar közkedveltté vált az IBM 5100-as rendszere, amely már perifériális felszereltségben is kitűnő volt. Hazánkban is számos példány található belőlük, és számos hívet (vagy hívőt?) szerzett az APL-nek.

A mai helyzetre leginkább a MicroAPL angol cég tevékenysége és termékei a jellemzőek. A cég mások által gyártott hardveren teljes APL-rendszereket szolgáltat, tokkal-vonóval-szoftverrel. Az egyik érdekes rendszere a SIGNET, amely Z80 és CP/M alapú. Érdemes egy pillantást vetni az összetételére:

- Z80 4 MHz-es mikroprocesszor,
- 64 kbájt RAM,
- 2 db RS232 kapu,
- 2 db 5 ¼" diszketta (2 × 390 kbájt),
- CP/M 2.2,
- APL 4.1,
- MicroSPAN oktatórendszer.

A fentiek együttvéve kb. 2500 dollárba kerülnek. Képernyővel, nyomtatóval mindez kb. 4000 dollár. Opcionálisan rendelkezésre áll még:

- MicroTASK, MicroFILE, MicroPLOT, MicroLINK, MicroFORM szoftverek,
- további 2 diszketta az eredeti házban,
- további soros kapuk,
- paralel kapuk,
- tetszőleges CP/M-kompatibilis szoftver.

SPECTRUM rendszerük 68000-es mikroprocesszorra épül. A rajta futó APL/68000 az amerikai Computer Companyvel közös vállalkozásban készült. Ez az APLSV bővített halmaza, fájlkezelő és egyéb segédprocesszorokkal felszerelve. Képességeire jellemző, hogy 6–8 egyidejű felhasználóval jobb válaszidőt és teljesítményt nyújt, mint a nagygépes időosztásos szolgáltatók átlagos terhelés mellett. Beépített Winchesterével a fájlkezelés teljesítménye is felveszi a versenyt a nagyokéval.

Mire jó?

A szerzői APL-t problémamegoldó rendszernek szánták; vagyis bonyolultabb, matematikai eszközöket igénylő feladatok megoldására. Nem véletlen ezek után, hogy az első években egyetemek, főiskolák, kutatóhelyek körében terjedt futótűzként a rendszer. Kutatáshoz, tervezéshez, fejlesztéshez a rendszer valóban ideális; a szokásos néhány heti programfejlesztési idő APL alkalmazásával két-három napra, nemegyszer órákra zsugorodik össze. A különbség valóban drámai! Az APL tömör szerkezete „nagypanelos" építkezése átmenetet jelent a nem procedúra-orientált alkalmazásfejlesztő rendszerek, mint a NOMAD, UFO, MANTIS, dBASE II stb. felé.

Miért is lehet ilyen gyorsan fejleszteni APL-ben? Ennek több összetevője is van. Maga a párbeszédes üzemmód a nyelvtől függetlenül is lehetővé teszi, hogy a gépet szinte folyamatosan, nagyon hatékonyan használjuk ki az emberi munka támogatására. Az APL nyelv tömör szerkezete 5–10-szeres forrásnyelvi programhossz-csökkenést jelent, ami elősegíti a jobb áttekintést. Az értelmező végrehajtás hibaleállás esetén lehetővé teszi a részletes állapotfelmérést, a korrekció vagy a hiba kijavítása után a futás folytatását (mivel a hibát nem gépi kódú utasításokban és hexadecimálisan kiírt adatok-

ban kell keresni, a hibabehatárolás meglepően gyors szokott lenni). Az APL-ben való munka több, mint élvezet: szellemi kihívás a kompilált programok sokszor robot jellegű javítgatásához képest, így a szubjektív feltételek is kedvezőbbek.

Hátránya viszont az APL-nek, hogy az elterjedt interpretív megvalósítás a végrehajtandó sorok állandó újrafordítása miatt sokkal lassabb futást eredményez, mint ugyanannak a feladatnak compiler-nyelven megírt programja. Emiatt rendszerint nem előnyös változatlan formában, rendszeresen futtatott programokat APL-ben megírni.

Meglepetésre, a fenti érvek ellenére a hetvenes években az APL számos, nem várt alkalmazási területen is előretört. Ezek közül néhány: az állami statisztika (a kanadai statisztikai hivatal gyakorlatilag csak APL-t használ); pénzügyi elemzés és tervezés, bankügyletek (az Egyesült Államokban a 70-es években az APL-rendszerek sorozatban nyertek el bankhálózat információrendszerére kiírt versenytárgyalásokat); szövegszerkesztés, szöveges információfeldolgozás; elemi iskolai oktatás (Anglia, Egyesült Államok); bonyolult ipari termékek gyártásának nyomon követése, minőségellenőrzés (például kozmetikai cikkek gyártásának diszpécser-rendszere).

Mit lehet erre mondani? A felhasználó mindent jobban tud…

Kinek jó?

Talán az APL volt az első számítógépes nyelv, amelynél kézzelfogható volt, hogy nemcsak egyfajta alkalmazásoknak, hanem egyfajta mentalitásnak, alkalmazói észjárásnak is kedvez.

Hosszú időn keresztül az operációs rendszerek, programcsomagok készítői magabiztosan hirdették, hogy a felhasználó ilyen meg olyan. Ezt a szemléletet ideje már a múltnak átadni. A felhasználók rendkívül sokfélék; sokfajta előélettel, előszeretettel és előítélettel; sokfajta vonzalommal, észjárással, és nem utolsósorban: céllal. (Talán nem is annyira meglepő ez, hiszen emberekről van szó!) Itt is hasznosíthatónak látszik a professzionális kereskedők megközelítése: a piac szegmentálása, azaz többékevésbé már tipizálható csoportokra bontása a mérhető tények alapján. Itt az idő a szoftverpiac professzionalizálódására, némi piacérzékenység megtanulására.

Nos, kinek is jó az APL? Felméréseim nincsenek, de tapasztalatom szerint azok szeretik az APL-t, akik hajlamosak (vagy inkább rákényszerülnek) elvontabb, szimbolikus gondolkodásmódra; vagy kézzelfoghatóbban: azok, akik szakmájukban az alkalmazott matematika (különösen az algebra, azon belül is a lineáris algebra) formalizmusát hasznosítják. Így egyértelműen szeretik az APL-t a mérnökök, a fizikusok, a csillagászok, a nyelvészek, a statisztikusok, a pénzügyi szakemberek, a biztosítók elemzői stb. (Egyébként is szerintem a matematika sok évszázados sikersorozata a valóság modellezésében éppen sajátos, nem verbális szimbolikájára, hallatlanul tömör nyelvére vezethető vissza.)

Úgy vélem, némi átgondolás után a fentiek többé-kevésbé elfogadhatók. Na, de miért vannak fanatikus APL-hívő orvosok, archeológusok, tanítók és politológusok? Talán azért, mert jó nyelv…

SÁROSSY JÓZSEF

(Vége)

PROGRAMOZÓI FOGÁSOK

Sorozatunkban ötleteket, módszereket adunk közre azok számára, akik már megismerkedtek mikrogépükkel, és/vagy azt munkájukban is használják. Reméljük, hogy e sorozat írásába ők is bekapcsolódnak; beküldött programozói fogásaikat – amennyiben valóban ötletesek – szívesen közöljük.

BASIC interpreterek – a tár szervezése

A kisebb, operációs rendszerek nélküli mikrogépek általában egy beépített BASIC interpretert és egy vele szorosan együttműködő szövegszerkesztőt tartalmaznak. Sorozatunkban először ezek működéséről, lehetőségeik kihasználásáról lesz szó. Ezen belül is a társszervezés módszereit vizsgáljuk meg, aminek ismerete nemcsak a gépi kódú alprogramok alkalmazásához elengedhetetlen, hanem lehetővé teszi a BASIC programok módosítását, szerkesztését, a ROM-ban tárolt EDITOR megkerülésével. Így például előállíthatunk olyan, amúgy jól működő BASIC programot, amely nem listázható. Mód nyílik az interpreter vagy a szerkesztő által nem megvalósított funkciók „kiváltására". Mondanivalónkat a hazánkban legelterjedtebb mikrogépek példáival illusztráljuk.

Az írható/olvasható memória (RAM) durván az alábbi részekre osztható:

(i) képernyő-memória ('k)
(ii) operációs rendszer munkaterülete
(iii) BASIC rendszerváltozók
(iv) BASIC munkaterület

Az (i)–(iii) alatt felsoroltak a legtöbb esetben a tárban rögzített helyen találhatók. A BASIC munkaterület nagysága eleve függ a mikroszámítógép kiépítettségétől, azonkívül a munkaterület több „darabban" is elhelyezhető a tárban.

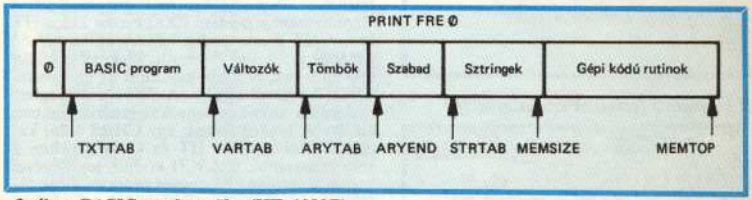
Egy – csak a BASIC munkaterületre vonatkozó – finomabb, de még mindig sematikus ábrázolás az *1. ábrán* látható.

A szerkesztés befejezésekor a dinamikus munkaterület gyakorlatilag üres. A program futása közben itt helyezi el az interpreter az egyes változók, tömbök értékeit, a GOSUB, FOR utasítások végrehajtásához szükséges információt. A C–64 *(2. ábra)* és a HT gépek *(3. ábra)* hasonló struktúrát mutatnak, aminek az az oka, hogy mindkettő Microsoft-típusú interpretert használ. (A mutatók nevei a Microsoft terminológiát követik.)

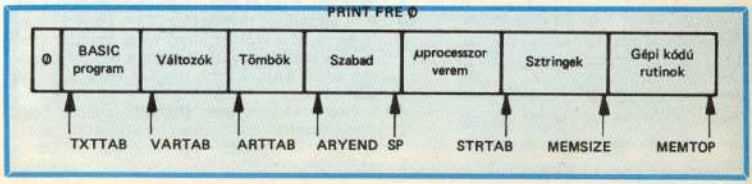
A ZX81 *(4. ábra)* ennél bonyolultabb, egy-

1. ábra. BASIC munkaterület

2. ábra. BASIC munkaterület (Commodore 64)



3. ábra. BASIC munkaterület (HT–1080Z)



részt mert külön vermet szervez a GOSUB végrehajtásához szükséges adatok tárolására, másrészt mert a BASIC munkaterület közepén található a képernyő-memória. Ennek egyetlen célja az, hogy a ZX81 1 k-s RAM-mal is forgalmazható legyen.

Az egyes mutatókat az alábbi címeken találhatjuk meg:

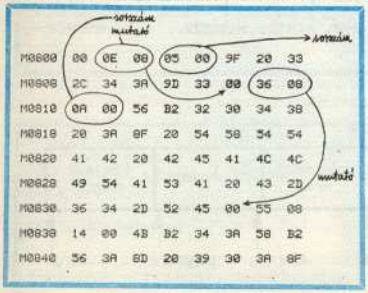
	ZX81	C–64	HT
TXTTAB	16509	43	16548
DSPTAB	16396	–	–
VARTAB	16400	45	16633
ARYTAB	–	47	16635
ARYEND	–	49	16637
WRKTAB	16404	–	–
STKTAB	16410	–	–
STKEND	16412	–	–
ERRSP	16386	–	–
STRTAB	–	51	16544
MEMSIZ	16388	55	16561
MEMTOP	kiépítéstől függ	40960	kiépítéstől függ
SP	µ processzor stack pointer (2 bájt)	(1 bájt)	(2 bájt)

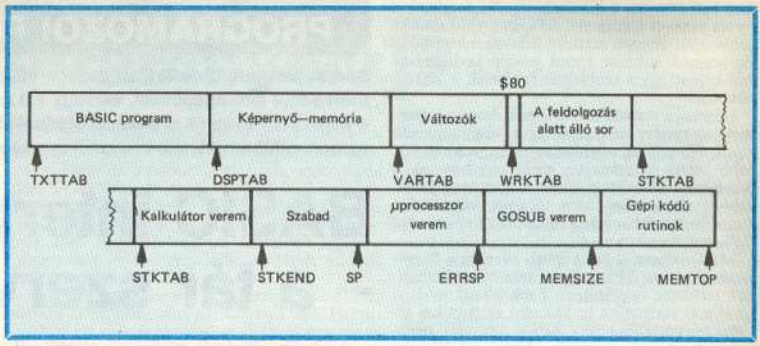
Ha már tudjuk, hol tárolja a BASIC interpreter a programot, könnyen készíthetünk olyan programrészt *(1. program)*, amelynek segítségével a program – abban a formában, ahogy a

1. program

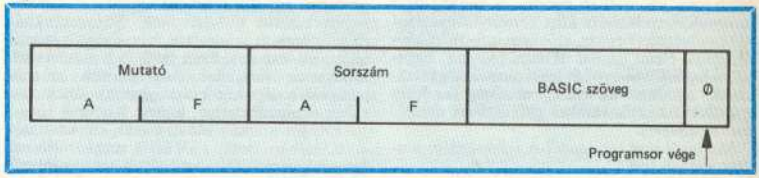
```
10 V=2048 : REM TXTTAB BEALLITASA C-64-RE
110 K=4:X=V:GOSUB 101 0: REM CIM KIIRASA
210 PRINT"M"+L$+" ";:K=2
310 FOR I=1 TO 8    :REM 8 EGYMAS UTANI
410 X=PEEK(V):V=V+1 :REM  MEMORIA KIIRASA
510 GOSUB 1010:PRINTL$+" "
610 NEXT I
710 PRINT
810 GET A$:IF A$="" THEN 81 0:REM FOLYTATAS
910 GOTO 110
1010 L=X:L$=""    :REM X HEXADECIMALIS ALAKBAN
1110 FOR J=1 TO K :REM L$-BA KERUL
1210 LX=L-INT(L/16)*16 : REM K-HOSSZAN
1310 L=INT(L/16)
1410 L$=CHR$(48+LX-(LX>9)*7)+L$
1510 NEXT J
1610 RETURN
```

Az 1. program futásának eredménylistája

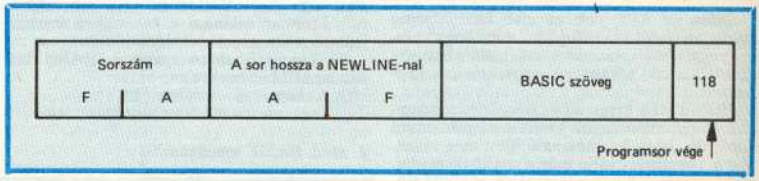




4. ábra. BASIC munkaterület (ZX81)



5. ábra. Egy programsor tárolása HT és C–64 gépeken



6. ábra. Egy programsor tárolása ZX81-en

memóriában tárolva van – a képernyőre kilistázható. (Ez nem azonos a LIST kiadásával. A továbbiak megértésére a programot úgy írtuk meg, hogy hexadecimális alakban írja ki az egyes memóriahelyek tartalmát. A programban elhelyezett megjegyzések elegendőek megértéséhez.)

A BASIC interpreterek az alapszavakat egyetlen bájton tárolják. Ennek a bájtnak az értékét az alapszó tokenjének hívják. A FOR alapszó tokenje például ZX81 esetén 235, a HT és a C–64 esetén pedig 129. A HT és C–64 esetében az =, +, −, * stb. jeleknek is külön tokenjük van. A többi karaktert az interpreter ASCII kódjával tárolja. A ZX81 a szövegben előforduló számkonstansokat bináris alakban tárolja. A bináris formát egy CHR$ (126) karakter előzi meg. A HT és C–64 gépeken a számkonstansok is ASCII kódjuk segítségével kerülnek tárolásra. Az egyes programsorok tárolását az *5. és 6. ábrán* foglaltuk össze. (A BASIC program végét további két Ø bájt jelzi. A program így az első programsor mutatójának Ø-ra állításával törölhető.)

Az A, F a szóban forgó egész számok felső és alsó bájtjait jelenti. Például a hexadecimálisan adott $AØB9 felső bájtja $AØ, alsó bájtja $B9.

Ennyi ismeret már elegendő ahhoz, hogy olyan utasításokat is megvalósíthassunk, amelyeket az interpreter – a priori – nem tud kezelni.

Kiszámított GOTO

A legtöbb BASIC interpreter nem engedi meg a GOTO X alakú utasítást, ahol X helyén egy tetszőleges aritmetikai kifejezés állhat (csak a GOTO 100 alak megengedett).

A *2. program* mutatja, hogyan valósítható ez meg. Az 1000-nél kezdődő programrész X értékének megfelelő stringet a GOTO ABCDE utasításban az ABCDE helyére másolja. Amikor a RETURN végrehajtódik, az ABCDE helyén már egy sorszám áll, és a program hibátlanul folytatódik. Ehhez természetesen ismerni kell a CIM értékét, amely a tár A-t (GOTO utáni első jelet) tartalmazó bájtjára mutat. Ha több helyről akarjuk a kiszámított GOTO-t, akkor az 1000-es alprogram meghívása előtt CIM értékét is állítanunk kell. (Valamelyik számban még visszatérünk arra, hogyan lehet a kiszámított GOTO-t a BASIC interpreter módosításával végrehajtani.)

Renew

A NEW BASIC parancs általában nem törli a memóriát, csupán néhány mutatót állít át, amiből az interpreter úgy érzékeli, hogy a BASIC program egyetlen sort sem tartalmaz. A ZX81 ilyenkor a képernyő-memóriát is mozgatja, ezért a program fizikailag is elvész. A C–64 és a HT gépek esetén azonban a program megvan, csak az első programsor mutatója Ø-ra lett állítva. A mutató értékét a POKE uta-

sítással helyre tudjuk állítani, feltéve, hogy tudjuk, hol kezdődik a következő sor. Ha nem, akkor a PEEK utasítás parancsként való használatával meg kell keresni az első Ø bájtot:

```
FOR I=TXTTAB+2 TO TXTTAB+258 :
IF PEEK(I)<>Ø THEN NEXT
```

Ezt követően a PRINT I+1 parancs kiírja az első programsor elején álló mutató helyes értékét. Ennek kiszámítjuk az alsó és felső bájtját, majd ezeket a TXTTAB és TXTTAB+1 címekre a POKE utasítás segítségével elhelyezzük. Hátra van még a VARTAB helyreállítása. A program végét jelző három Ø bájtot a következő parancs segítségével találhatjuk meg:

```
FOR I=TXTTAB TO 38ØØØ :
IF PEEK(I) + PEEK (I+1) +
PEEK(I+2)<>Ø
THEN NEXT
```

Ezután a PRINT I+3 parancs kiírja a VARTAB mutató helyes értékét, amit a TXTTAB-nál ismertetett módon átállíthatunk. (Megjegyezzük, hogy a változók, tömbök értékeit ezzel a módszerrel nem állíthatjuk vissza. Miért?!)

Append

Gyakran szükség van arra, hogy programírás közben a programhoz további, már meglevő programrészleteket illesszünk hozzá. Ezt a LOAD, CLOAD stb. parancsok nem teszik meg, hiszen felülírják a meglevő programot. A TXTTAB átállításával azonban elérhetjük, hogy a töltőparancs kiadása után ne íródjék felül az eredeti program.

A ZX81 esetében ez nehézkes, hiszen a SAVE utasítás az összes rendszerváltozót kimenti, így a visszatöltésnél a TXTTAB értéke átállítódik. HT és C–64 gépek esetén ez a módszer működik. Az APPEND utasítás a következőképpen hajtható végre:

(i) A TXTTAB mutató értékét VARTAB–2-re állítjuk. (A program végét jelző két Ø bájtot töröljük.)

(ii) Betöltjük az első programhoz hozzáfűzni kívánt programot.

(iii) A TXTTAB mutató értékét visszaállítjuk a normális értékre.

(iv) A hozzáfűzött sor elején álló mutatók elromlottak. Ezt a POKE, PEEK utasításokkal igen fáradságos helyreállítani. Két eset lehetséges. Vagy tudjuk, hol a ROM-ban az a rutin, amely ezt az újraszámozást elvégzi, vagy az első programnak kell egy olyan rutint tartalmaznia, amely elvégzi ezt az újraszámozást.

Parancs módban ezek a lépések minden nehézség nélkül végrehajthatók. Az eljárás támogatására BASIC program csak igen nehézkesen készíthető, hiszen a TXTTAB és VARTAB mutatók átállítása, illetve a töltés befolyásolja a BASIC interpreter működését.

Az APPEND utasítás végrehajtását támogató gépi kódú programot *(3. program)* mutatunk be utolsó példaként. A programot C–64-re készítettük el, de HT gépre is – hasonló elvek alapján – elkészíthetjük. A programot a kívánt szerkesztések végrehajtása előtt be kell töltenünk a memóriába. Az APPEND utasítás hatását a következő paranccsal érhetjük el:

SYS 12• 4096, „file név", <egységszám>.

Például a SYS 12• 4096, „KUKAC" a KUKAC nevű szalagon, a SYS 12• 4096, „RUTINOK", 8 a lemezen levő RUTINOK nevű fájlt fűzi hozzá a programhoz. A magyarázatokat az assembler listán adtuk meg.

DR. ÚRY LÁSZLÓ

```
10 INPUT X                    : REM A VEZERLES X. SORBA KERUL
20 GOSUB 1000: GOTO AAAAA     : REM KISZAMITOTT GOTO UTASITAS
30 PRINT30:STOP
40 PRINT40:STOP
530 PRINT530:STOP
990 REM A 20.SORBAN AZ AAAAA HELYERE STR$(X) KERUL.
995 REM A VISSZATERES UTAN AAAAA HELYEN MAR CIM VAN
1000 X$=STR$(X):CIM=8*256+23:IF X>40000 OR LEN(X$)>6THEN 600000
1010 FOR I=1 TO LEN(X$)-1 :REM X$-T KARAKTERENKENT
1020 POKE CIM+I-1,ASC(MID$(X$,I+1,1)) :REM ATMASOLJUK
1030 NEXT I
1040 IF LEN(X$)=6 THEN RETURN
1050 FOR I=LEN(X$) TO 5
1060 POKE CIM+I-1,32
1065 NEXT
1070 RETURN
11111 PRINT11111:STOP
```

2. program

3. program

```
C000                    10          *=$C000         ; A PROGRAM KEZDETE
0277                    20 KEYBUF   =$0277
00FB                    30 CIM      =251
C000                    40          ;
C000                    50          ; A BASIC KEZDET MUTATO ATALLITASA
C000                    60          ;
C000 38                 70          SEC
C001 A5 2D              80          LDA 45          ; (TXTTAB)=(VARTAB)-2
C003 E9 02              90          SBC #$02
C005 85 2B             100          STA 43
C007 A5 2E             110          LDA 46
C009 E9 00             120          SBC #$00        ; AZ ATVITEL LEVONASA
C00B 85 2C             130          STA 44
C00D                   140          ;
C00D                   150          ; A SYS 49186 [RETURN] UZENET
C00D                   160          ; ATMASOLASA A PUFFERBE
C00D                   170          ;
C00D A2 00             180          LDX #$00        ; X A PUFFER ELEJERE MUTAT
C00F BD 54 C0          190 LP2      LDA SZOV1,X
C012 9D 77 02          200          STA KEYBUF,X ; EGY KARAKTER ATIRASA
C015 E8                210          INX
C016 E0 09             220          CPX#$09
C018 D0 F5             230          BNE LP2         ; UGRAS, HA VAN MEG KARAKTER
C01A 86 C6             240          STX $00C6       ; X DB. BILLENTYU VAN A PUFFERBEN
C01C 20 73 00          250          JSR $0073       ; , ATLEPESE
C01F 20 68 E1          260          JSR $E168       ; A PROGRAM BETOLTESE
C022                   270          ;
C022                   280          ; (VARTAB) JO HELYRE MUTAT, DE
C022                   290          ; A PROGRAM VEGET JELZO KET Ø BYTE HIANYZIK
C022                   300          ;
C022 A2 00             310          LDX #$00        ; A BASIC ELEJE MUTATO
C024 A9 01             320          LDA #$01        ; HELYREALLITASA.
C026 85 2B             330          STA 43          ; (TXTTAB)=$0801
C028 A9 08             340          LDA #$08
C02A 85 2C             350          STA 44
C02C 38                360          SEC
C02D A5 2D             370          LDA 45          ; A (VARTAB)-2 ERTEK
C02F E9 02             380          SBC #$02        ; BETOLTESE A CIM-RE
C031 85 FB             390          STA CIM
C033 A5 2E             400          LDA 46
C035 E9 00             410          SBC #$0
C037 85 FC             420          STA CIM+1
C039 A9 00             430          LDA #$0
C03B 81 FB             440          STA (CIM,X) ; A Ø BYTE BEIRASA
C03D A5 2D             450          LDA 45
C03F 18                460          CLC             ; A (VARTAB)-1 MUTATO
C040 A5 FB             470          LDA CIM         ; CIM-RE VALO BETOLTESE
C042 69 01             480          ADC #$1
C044 85 FB             490          STA CIM
C046 A5 FC             500          LDA CIM+1
C048 69 00             510          ADC #$0
C04A 85 FC             520          STA CIM+1
C04C A9 00             530          LDA #$0
C04E 81 FB             540          STA (CIM,X) ; A MASIK Ø BYTE BEIRASA
C050                   550          ;
C050                   560          ;A MUTATOK HELYRE ALLITASA
C050                   570          ;
C050 20 33 A5          580          JSR $A533
C053 60                590          RTS
C054 53 59 53          600 SZOV1    .TEXT "SYS49186"
C05C 0D                610          .BYTE 13    ;SYS 49186 [RETURN]
C05D                   620          .END
```

OKISZ Szervezési és Számítástechnikai Vállalat
Budapest 8. Pf.: 247
Telefon: 340-102

OKISZ Szervezési és Számítástechnikai Vállalat

Szoftver termékek és számítástechnikai szolgáltatások kis- és középüzemek részére

Ügyvitelszervezés és gépesítés
A ROBOTRON 1372, 1355, és 1711 típusú gépekre kidolgozott és nagy számban telepített rendszereink:
- anyagkönyvelés
- költségvetés készítés
- számlázás
- fogyóeszköz könyvelés
- főkönyvi és folyószámla-könyvelés
- bérszámfejtés, bérügyvitel
- készárukönyvelés
- gyártáselőkészítés
- utókalkuláció

A több száz alkalmazói rendszerből a feladatoknak, az igényeknek megfelelő referencia helyet tudunk bemutatni.
Az ügyvitelgépesítésnél „kulcsrakész" feladat ellátását biztosítjuk, mely kiterjed a gépek beszerzésére, a segédanyagok biztosítására, oktatásra és betanításra is.

Ügyvitelszervezési Osztály **Tel.: 336-399**

Mikroszámítógépes alkalmazások
A VIDEOTON VT 20 és VT 30, a ROBOTRON 5100-as gépcsalád és FLOPPYMAT SP mikroszámítógépek alkalmazására kidolgozott típusrendszereink lehetőséget adnak a fokozatos alkalmazásfejlesztés megvalósítására:
- anyag-, áruforgalmi rendszer
- készletgazdálkodási rendszer
- termelést előkészítő és támogató szolgáltatások
- kereskedelmi információs rendszer

Szervezési Főosztály **Tel.: 426-936**

Számítógépes feldolgozások
Saját fejlesztésű rendszerek teljes körű feldolgozásait biztosítjuk kötegelt üzemmódban, adatrögzítéssel és adatkonvertálással.
A feldolgozások országos alkalmazási lehetőségét a SZÜV megyei számítóközpontjaival együttműködve valósítjuk meg.
Vállalkozunk egyedi feldolgozási rendszerek kifejlesztésére, mintarendszereink adaptálására, applikálására:
- anyaggazdálkodási általános rendszer
- építőipari anyaggazdálkodási rendszer
- termelésirányítást támogató rendszer
- ipari szövetkezeti irányítási rendszer
- áruforgalmi rendszer

Számítóközpont **Tel.: 210-808**

Ajánlunk:

Kész rendszereket,
adaptációval, vagy anélkül
tanácsadást
fővállalkozást
egyedi rendszerek fejlesztését

FLOPPYMAT–SP mikroszámítógépre kész rendszereinket

- az ipar
- a kereskedelem
- a mezőgazdaság és
- az államigazgatás

területére dolgoztuk ki, például:
- bolti elszámoltatás
- anyagkészlet-nyilvántartás és elszámolás
- személyzeti nyilvántartás
- bérszámfejtés
- vevői, ill. szállítói rendelés-nyilvántartás funkciókra

Bemutatót, ismertetést vállalunk

tagvállalatainknál
referencia cégeknél
képviseleti irodánkban, ill.
kívánságra
az érdeklődő telephelyén is.

SZERVEZ
GÉPET INSTALLÁL
BÉRLETEZ
RENDSZERT KÉSZÍT ÉS BETANÍT

FLOPPYMAT

FEJLESZTŐ
GAZDASÁGI
TÁRSASÁG

Tagvállalatok

Vegyipari Számítástechnikai Fejlesztési Társulás
Képviseleti Iroda:
1063 Budapest,
Szinyei Merse u. 1.
tel.: 127-628

SZÖVORG
Szövetkezeti Gazdaságszervezési és Számítástechnikai Iroda
1028 Budapest, Attila u. 13.
tel.: 189-162

ügyintéző:
Sólyom József főosztályvezető

KERSZI
Kereskedelmi Szervezési Intézet
1134 Budapest,
Dózsa György út 150.
tel.: 858-579 202-650

ügyintéző:
Szigeth Gábor osztályvezető

MÜSZI
Mezőgazdasági Ügyvitelszervezési Iroda
1021 Budapest,
Vörös Hadsereg útja 95.
tel.: 858-579

ügyintéző:
Kaszap László főosztályvezető

VILATI
Villamos Automatika Fővállalkozó és Gyártó Vállalat
1066 Budapest, Ó utca 27.
tel.: 122-866

ügyintéző:
Szabó József főosztályvezető

Elektronikus feladatlap VU–CALC ZX-Spectrumra

Amikor 1979-ben a VisiCorp cég megjelent a piacon a pénzügyi tervezést és modellezést támogató, ún. electronic spreadsheet-tel, még nem gondolta, hogy ebből a programcsomagból rövid idő alatt 500 ezer példányt fog eladni.

A VisiCalc – ez volt a termék neve – titka pedig elég egyszerűnek látszott: nagyfokú felhasználóközeliség, a közvetlen manipuláció elvének („what you see is what you get") alkalmazása és a vizuális programozás.

A siker a többi szoftvergyártót is az ilyen jellegű programcsomagok készítésére ösztönözte.

A PSION cég 1981-ben piacra dobta a ZX-Spectrumra kifejlesztett – a VisiCalc-hoz nagyon hasonló – programcsomagját, a VU–CALC-ot. A VU–CALC 16 k-s kiépítésnél 18 × 13, 48 k-s kiépítés esetén 60 × 60 elemből álló táblázatot tud kezelni.

Közvetlenül a betöltés után a képernyő elrendezése a következő: kétsoros parancsmező a képernyő tetején, egy input sor a képernyő alján és egy üres terület középen, amelyben mozgatható ablak helyezkedik el. Az ablakon a táblázatnak egy 18 × 4 elemből álló részlete látható. A táblázat sorainak azonosítása felülről lefelé haladva, alfabetikusan történik, az oszlopok számozása pedig a felső részen van elhelyezve, balról jobbra emelkedő sorrendben. Ennek megfelelően a kurrens oszlopszámok mindig közvetlenül az ablak felett láthatók.

A táblázat elemeit boxoknak nevezzük. Egy box helye két koordinátájával egyértelműen meghatározható. Van egy speciális box, amelyet egy vörös téglalap illusztrál. Ez a cursor-box, amely szabadon mozgatható az ablakon – a klaviatúrán lévő 4 cursor-billentyű segítségével. Abban a szélső helyzetben, amikor a cursor eléri az ablakon lévő utolsó sort vagy oszlopot, és még mindig nyomjuk a billentyűt, az ablak halad tovább. Ennek segítségével az egész táblázatot bármely irányban gyorsan tudjuk mozgatni.

A cursor-boxon keresztül lehet bevinni adatot, szöveget vagy formulát. Az egyes boxokba numerikus és string jellegű adatokat, valamint formulákat írhatunk be.

A numerikus adatok megadása egyszerűen az alsó input sorba való begépeléssel történik. Az ENTER gomb megnyomása után az adat beíródik abba a boxba, ahol a cursor áll. Stringek megadása hasonlóan történik, az egyetlen különbség, hogy stringet mindig idézőjellel kell kezdeni.

A numerikus adatok maximális hossza 7, a stringeké 28 karakter.

A formulákat szintén az alsó input sorba kell

1. kép

beírni. Az ENTER gomb megnyomására a kurrens boxban azonnal megjelenik a kiszámított érték, az input sor és az ablak között pedig az adott formula. A formulák tartalmazhatnak konstansokat, boxokra való utalást, valamint egyszerű aritmetikai műveleti jeleket: +, −, *, /. A boxokra való hivatkozás a sor- és oszlopindex segítségével történik. A VU–CALC csak inicializált hivatkozásokat fogad el. Példa a formulákra: D12 × (B2 + 1.5)/C1.

További lehetőségként olyan speciális formulákat is be lehet vinni, amelyek meghatározott boxokat összegezni tudnak. Az összegezés történhet egy soron vagy oszlopon, valamint egymás mellett, illetve alatt álló boxok téglalap alakú blokkján. Az utasítás formája: & f:l. Az f paraméter a blokk bal felső boxa, az l pedig a jobb alsó box. Például & A2:B4 jelentése:

A2 + A3 + A4 + B2 + B3 + B4.

A C sorban való összegzés például: & C3:C5 = C3 + C4 + C5. Az f boxnak mindig az l boxtól balra és felette, egyetlen sor, illetve oszlop esetén pedig csak balra, illetve csak felette szabad lennie.

A boxokban egyaránt tárolásra kerül az érték és a formula, de a formula csak akkor jelenik meg, ha a cursort az illető boxra helyezzük.

Ha az input sorba # jelet írunk, a VU–CALC átmegy parancs módba, és a parancsok listája megjelenik a felső két sorban (a parancsmezőben). A kiválasztott parancs az első karakterének és a szükséges paramétereknek a beírása után végrehajtásra kerül.

A parancsok és jelentésük a következő:

B (Blank): Blank-kal tölti fel azt a boxot, ahol a cursor áll.

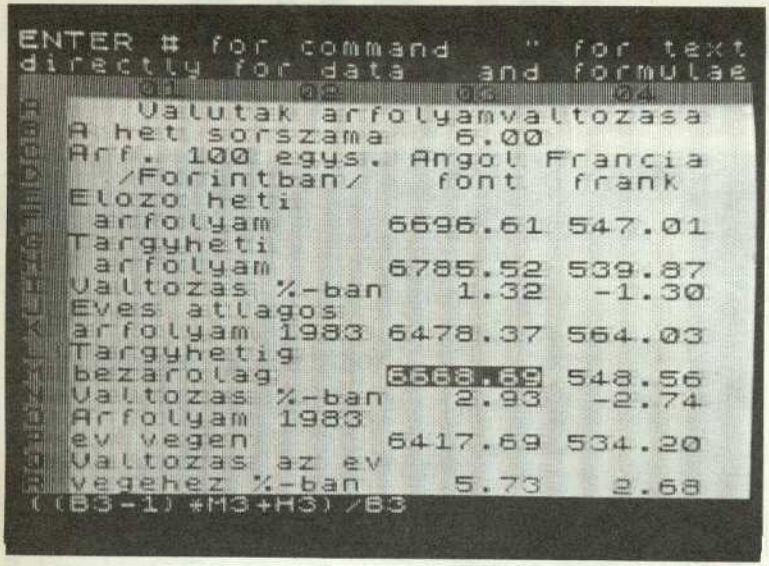
F, c, f, j, (Format): Adott oszlopban (vagy oszlopokban) lévő számok ábrázolásának megadására szolgál. c értéke lehet egy oszlopszám vagy „A". Az első esetben az ábrázolás csak a megadott oszlopra vonatkozik, a második esetben az egész táblázatra. Az f paraméter az ábrázolás típusát adja meg, ami lehet egész, ha f = I, valós, ha f = $ és ún. általános alak, ha f = G (generál). A j paraméter a boxon belüli elhelyezkedést határozza meg. Ha j = L, akkor a szám balra, ha J = R, akkor jobbra igazodik.

G, rc (Go): Közvetlen ugrás az rc boxra.

E (Edit): A formulák szerkesztését teszi lehetővé.

C (Calculate): A teljes táblázatot újrakalkulálja, a megváltoztatott adatok, illetve formulák alapján. Mivel a VU–CALC az eredetileg bevitt vagy újraszerkesztett formulák sorrendjét nem őrzi meg, a # C parancsot minden formula változtatása után ki kell adni.

T, r vagy c, r vagy c (Transfer): Ez az utasítás az első paraméterrel meghatározott sort vagy oszlopot a második paraméterrel meghatározott sorba vagy oszlopba viszi át. Sort nem másol át oszlopba, sem oszlopot sorba.



2. kép

R, rc, f:l (Repeat): A parancs egy box értékének vagy a boxban lévő formulának az ismétlését teszi lehetővé. Az első paraméter rc, egy boxra való hivatkozás, amely box értékét vagy formuláját ismételni akarjuk. Az ismétlés történhet egy soron vagy oszlopon, valamint egymás mellett, illetve alatt álló boxok téglalap alakú blokkján. Ezt az f:l paraméterek határozzák meg, ahol f a blokk bal felső boxa, l pedig a blokk jobb alsó boxa. Például: A3:B5 esetén az ismétlés a következő boxokra vonatkozik: A3, A4, A5, B3, B4, B5. Szükséges feltétel, hogy az f boxnak mindig az l boxtól balra és felette, egyetlen sor, illetve oszlop esetén csak balra, illetve csak felette szabad lennie.

Q (Quit): Engedélyezi a munkaterület törlését vagy a programból való kilépést.

S (Save): Törli a képernyőt, és a táblázatot a megadott néven, kazettán tárolja.

L (Load): Törli a képernyőt, és betölti a megadott nevű táblázatot.

P (Print): A képernyőről az outputot kiírja nyomtatóra.

Az eddigiekből is látható, hogy a VU–CALC elsősorban olyan feladatok megoldásában nyújt hasznos segítséget, amelyeknek változtatható paramétereik vannak, hiszen egyetlen paranccsal (Calculate) regisztrálni tudjuk az egész táblázat változását.

Alkalmazási példa

A táblázat *(1. kép)* aktualizálása a következő módon történik:

a) A tárgyheti árfolyamok áthelyezése az „előző heti árfolyam" sorba: # T, h, f.

b) A tárgyheti árfolyamokban bekövetkezett változások begépelése és a hetek sorszámának módosítása.

c) Újrakalkulálás: # C

A fenti módosításokkal, illetve parancsok kiadásával nemcsak az ablak tartalma változott meg, hanem az egész táblázat újra kiszámításra került *(2. kép)*. Ugyanis az ablak a maximálisan 60 × 60-as méretű táblázatnak csak ötvened része. Az ablak mozgatásával a nem látható részekbe is „betekintést" nyerhetünk.

A példa igen egyszerű, csak a bemutatást szolgálja, hiszen egy számítási lépcsőben jutunk el a feladat megoldásához.

Olyan feladatok (például optimalizálás) esetében, ahol sok lépcsőben, többszöri változtatással és a próba – szerencse módszer alkalmazásával jut el a felhasználó a végső megoldáshoz, a VU–CALC által támogatott electronic spreadsheet nélkülözhetetlen segédeszközzé válik.

A Betonútépítő Vállalat – amelynek tulajdonában van ez a programcsomag – a következő alkalmazásokat tervezi: pénzügyi szimulációk, bérelszámolás, számlázás, szerződés-nyilvántartás, árajánlatok készítése.

Természetesen ez még akkor is csak egy töredéke azoknak a feladatoknak, amelyekre a VU–CALC és a hasonló szoftvereszközök a gyakorlatban eredményesen használhatók lesznek, ha figyelembe vesszük, hogy a prognózisok szerint a mikroszámítógépek alkalmazásában a leglátványosabb fejlődés a vállalati irányítás és gazdálkodás területén várható.

MÓCZÓ JÓZSEF

A Floppymat családhoz tartozó berendezéseket a helyi feldolgozási igények, így az adatrögzítés, a számítógéppel támogatott decentralizált ügyvitel és adatfeldolgozás, valamint a központi feldolgozó rendszerekkel való sokoldalú együttműködés támogatására alakította ki a VILATI. Ennek megfelelően nem személyi számítógépnek, hanem inkább irodai-ügyviteli gépnek tekinthetők.

A hardver

A család egyes modelljeinek központi perifériális eszköze a berendezések nevében is szereplő hajlékony mágneslemez (floppy). Az alkalmazott MOM MF 6400 típusú meghajtók (max. 4 db) 8"-es, egyoldalas és dupla sűrűségű lemezek kezelését teszik lehetővé, de a Floppymatok csak szimpla sűrűségű írást és olvasást engednek meg (250 kbájt lemezenként). A lemezek formátuma IBM szoft szektoros, így azok csereszabatosak más gépek azonos méretű floppyjainak formátumával.

A berendezéseket Fairchild F8 típusú, 8 bites mikroprocesszor vezérli. Ez a választás hardver szempontból számtalan előnnyel jár a gyártó számára, mivel az ismert mikroprocesszorokkal ellentétben az F8 áramköri család chip-készlete olyan kialakítású, hogy az egyes chipek egyszerre több funkciót is tartalmaznak. Így például a processzor chip 2 db I/0 porttal és 64 bájt RAM-mal is el van látva. Ezzel a számítógéphez szükséges elemek száma jelentős mértékben csökkenthető. Emiatt viszont az utasításkészlet kialakításakor a Fairchild kevésbé tudta annak idején (1976) figyelembe venni a programozás szempontjait.

A modulárisan bővíthető számítógépcsalád alsóbb tagjai kisebb tárkapacitásuk következtében (12 kbájt ROM, 4 kbájt RAM) csak a gyártó által szállított alkalmazói programokkal (FMN06/08) használhatók.

A Floppymat–E egy, a Floppymat–I két meghajtót tartalmaz. A Floppyline–E és a Floppyline–I 16 bites, TTL szintű I/0 csatornával bővítik az E, illetve I modelleket. A csatorna segítségével eddig megvalósított illesztések a TPA 70 és PC 4000, VILATI gyártmányú miniszámítógépekkel biztosítják a gyors, közvetlen kapcsolatot, illetve szabványos BSI csatlakozási felületet nyújtanak.

A Floppymat–SP 60 kbájt kapacitású RAM tára (+ 2 kbájt ROM) teljes mértékben kihasználhatóvá teszi az adott mikroprocesszor lehetőségeit, és a gyártótól függetlenül, szabadon programozható.

A Floppymat–SPD az SP modell 2 db floppy meghajtója mellett azoknál lényegesen nagyobb kapacitású lemezegységekkel is rendelkezik. A hazai minigépeken jól ismert, IZOT SZM–5400 típusú meghajtóból max. 4 db kapcsolható a géphez. Egy meghajtó 2,5 Mbájtos fix lemezzel és egy ugyancsak 2,5 Mbájtos cserélhető, kazettás (cartridge) lemezzel rendelkezik.

A mikroszámítógép, a floppy meghajtók és a billentyűzet íróasztalszerű kialakításban egybeépült. A billentyűzet kombinált alfanumerikus, numerikus és funkcionális részekből áll. Kiala-

A Floppymat család

kítása valamennyi típus esetében azonos és jól áttekinthető. A funkcionális billentyűzet használatát külön feliratjelzések könnyítik. A billentyűzetet a felhasználó kívánságára a megfelelő magyar ékezetes karakterekkel is szállítja a gyártó. Ebben az esetben a Floppymat képernyője és a csatolt nyomtató is ismeri ezeket a karaktereket, illetve a később ismertetendő szoftverelemek is támogatják azokat. A hazai mikroszámítógépes termékek között ezzel a szolgáltatással (sajnálatos módon) egyedülállóak a Floppymatok.

A képernyő az asztal bal oldalán helyezkedik el. Mérete valamivel kisebb, mint az a személyi számítógépeknél általában megszokott (28 cmes képátló). Ezt a hátrányt ellensúlyozza az E és I modelleknél alkalmazott, kifejezetten nagy karakterméret (8 sor, soronként 32 karakter). Az SP modell esetében a képernyő funkcionalitását is növelték. 16 kisbetűs vagy 9 nagybetűs sorban max. 64, illetve 32 karakter jeleníthető meg egyszerre. A karakterek generálása 7×9 méretű pontmátrixban történik, a memória részét képező $1+1$ kbájt kapacitású tárból kivéve az érintett karakterek kódját. A képernyő programozása így igen hatékony és rugalmas.

A berendezésekhez MERA DZM–180 típusú, Zbrojovka Consul 2111 típusú és Epson MX–80 típusú (vagy azzal ekvivalens) mátrixnyomtatók csatlakoztathatók. Igény esetén lyukszalagos olvasó (FS 1501) és lyukasztó (PL150) berendezések is rendelhetők. Adathordozó-konvertálási célokra ECMA 34 kompatibilis digitális kazettaegység (típusa MERA PK–1) áll rendelkezésre.

A megvásárolható adatátviteli csatlakozás CCITT V24 kompatibilis, univerzális szinkron-aszinkron adatátviteli interfészen (USART) alapszik, félduplex üzemmódban max. 9600 Baud sebességű átvitelt tesz lehetővé.

A családhoz szorosan kapcsolódó Floppymat Konverter önálló berendezés. Ez a hajlékony mágneslemezek és a nagyobb számítógépeken használt, 800 bit/inch felírási sűrűségű, 9 csatornás mágnesszalagok között biztosítja a megfelelő adathordozó-konverziót.

A Floppymat gépcsalád nagyfokú modularitásával, az alkalmazható perifériális eszközök és adatátviteli kapcsolatok széles választékával hardver tekintetben igen jól megfelel a helyi feldolgozások hazai igényeinek.

Korlátozó tényezőként szinte csak a viszonylag kis kapacitású képernyőt, valamint a nem minden alkalmazáshoz megfelelő, robusztus, íróasztalszerű kivitelt említhetjük. (A gyártótól kapott tájékoztatás szerint, ezt felismerve, elkészült a nagyképernyős asztali változat. – Szerk. megj.) Kívánatos lenne a floppy meghajtók dupla sűrűségű felírásmódjának kihasználása is, ami a hajlékony mágneslemezek tárolási kapacitását a duplájára növelné. Megjegyezzük azonban, hogy ez más hazai gyártmányoknál is probléma (lásd például a TAP 34 ismertetést előző számunkban).

A rendszerszoftver

A Floppymatok F8 típusú mikroprocesszorának utasításkészlete alapvetően különbözik az elterjedt 8 bites mikroszámítógépekben alkalmazott típusoktól (Intel 8080, 8085, Zilog Z80, MOS Technology 650x, Motorola 680x). Így nem lehetett a széles körben használt operációs rendszereket, például a CP/M-et sem adaptálni. A gyártó saját operációs rendszerként az ún. EXDOS-t fejlesztette ki, amely jelenlegi változatában a hajlékony mágneslemezes konfiguráción végzett szoftverfejlesztést támogatja.

Az Assembler fordítóprogram az F8 utasításkészletnek megfelelő mnemonikus kódokat és az assembly nyelvben bevezetett adatleíró direktívákat ismeri fel, illetve a segítségükkel megfogalmazott forrásprogramból generálja a megfelelő bináris tárgykódot. Ennyiben tehát nem különbözik más assemblerektől. Az F8 meglehetősen leegyszerűsített utasításkészlete ugyanakkor sokkal munkaigényesebbé teszi az assembly szintű programozást, és még olyan programozóknak is nehézséget okozhat, akik már gyakorlottak az ilyen gépközeli programozásban (például az architektúra sok regiszter-tartalom-mozgatást igényel, viszont az ezt kezelő utasítások meglehetősen zavarba ejtőek).

A PLF8 fordítóprogram a gyártó saját fejlesztésű nyelvében leírt forrásprogramok megfelelő feldolgozására és tárgykódjának előállítására képes. A PLF8 nyelvet a gyártó Pascal subsetként hirdeti, azonban inkább nevezhetnénk azt a Pascal nyelv logikájából kiindulva a Floppymat adottságaihoz igazított, sajátosan leegyszerűsített, magas szintű nyelvnek. Számos meglepő korlátozása van; így például nem rendelkezik a Pascalra alapvetően jellemző formális paramétermegadás lehetőségével a deklarálható eljárásokban. A Pascalban meglehetősen nyitva hagyott bemenet-kimenet tekintetében ugyanakkor fontos szolgáltatásokkal rendelkezik. Hatékonyan támogatja az állomány-, rekord- és az utóbbin belül kötetlen karaktersorozat-kezelést. Tartalom és cím szerinti keresést tesz lehetővé floppys állományokon, véletlenszerű és direkt eléréssel. Így a floppy-orientált alkalmazásfejlesztéshez is megfelelő támogatást nyújt.

A Link nevű tárgykódszerkesztő program több fordítás eredményét szerkeszti össze futtatható gépi kódú programmá. A hajlékony mágneslemezen ilyen módon összeállított programok betöltését és futtatását a Manko nevű program támogatja. A nyomkövetéshez a Debug nevű program áll rendelkezésre. A forrásnyelvi programok szerkesztését az Editor segíti. Az operációs rendszer ezenkívül még a lemez másolásához és inicializálásához nyújt segédprogramokat.

Az EXDOS-tól független rendszerszoftver-támogatást biztosít a DIANA GM és a PROGRESS GM által Floppymatra adaptált ügyviteli Pascal rendszer. Ennek a más gépeken is használt rendszernek (lásd a TAP 34 ismertetést előző számunkban) lényeges sajátossága, hogy az EXDOS-szal ellentétben a kazettás lemezek kezelését is támogatja az SPD típuson. (A gyártótól kapott tájékoztatás értelmében ez a támogatás a közeli jövőben a PLF8-ban is megvalósul. – Szerk. megj.)

A Floppymatok rendszerszoftver-támogatása eléri azt a szintet, amit a jelenlegi felhasználási körülmények igényelnek. Az EXDOS meglehetősen gyártó-specifikus volta ugyanakkor jelentősen korlátozza korszerű szoftverelemekkel való bővíthetőségét, amire más rendszerek esetében külső forrásból is lehetőség nyílik.

Alkalmazásfejlesztési támogatás

A Floppymatok erőssége a piacon jelenleg kapható hazai gyártású személyi számítógépekkel szemben, hogy a gyors és közvetlen alkalmazásfejlesztéshez viszonylag fejlett eszközrendszerrel rendelkeznek.

Az FMN06/08 programcsomag olyan általános adatállomány-kezelő rendszer, amely egyszerű és mérsékelt bonyolultságú ügyviteli célokra kiválóan alkalmazható. Adatrögzítési funkciója max. 10 bizonylat kezeléséhez nyújt programozható támogatást. Gyűjtési funkciója hajlékony mágneslemezes állományokból 3 db, 19 bájtos regiszterben képes előjelhelyes összegzést végezni, szintén programozható módon. Az állományokon cím és tartalom szerinti gyors kezelést is biztosít.

Az FMN06/08 bővített funkciói a kétmeghajtós változatokon (I, SP, SPD) állnak rendelkezésre. A másolási funkció lemezek, állományok és rekordok másolását támogatja. Egy másik funkciócsoporttal rendezés és válogatás hajtható végre, az igényeknek megfelelően, programozott módon.

Az FMN06/08 programcsomag az IBM 3741 típusú adatrögzítő rendszerrel felülről kompatibilis szolgáltatásokat nyújt. Az alkalmazás programozása képernyőn keresztül végzett paraméterezéssel történik. A gyűjtésnél például logikai ÉS, VAGY és NEM kapcsolatok hozhatók létre állományok rekordjai és azon belüli tetszőleges karaktermezők között. A programcsomagot részben ROM-ba beépítve vagy lemezen szállítja a gyártó, és használata a billentyűzettel is megfelelően integrálva van.

Az előbbi programcsomag szolgáltatásait funkcionálisan jelentősen kibővíti a MIKRO MINI Software GM által a Vegyipari Számítástechnikai Fejlesztési Társulás megbízásából kifejlesztett UPDT/TABL programcsomag (forgalmazó a VSZFT). Ezzel a közepes bonyolultságú decentralizált ügyintézési és adatfeldolgozási alkalmazások is könnyen és hatékonyan fejleszthetők.

Az UPDT az általános állomány-karbantartást támogatja sokrétűen. Egy állományhoz több képernyőterv tartozhat, ernyőnként különböző tartalommal. A váltás közöttük egyszerű billentyűlenyomással történik. Képernyő-kommunikáció segítségével ugrás hajtható végre az állomány elejére, végére, illetve keresés kulcs (bármelyik mező) és tartalom (bármelyik karaktersorozat) szerint. Az utóbbi esetben a mezők és a mezőrészek között tetszőleges ÉS kapcsolatok definiálhatók a képernyőn keresztül.

A megjelenített kép billentyű lenyomásával kiíratható. További szolgáltatás a megjelenített rekordok közötti léptetés, adott szempontoknak megfelelően és adott sorrendben.

Új rekordokat szekvenciálisan lehet beszúrni. Módosítás hajtható végre bármely létező rekordra, ciklikusan ismételve egy akár több képernyőn megjelenített rekordon belül. Ennek során rugalmas, képernyőn belüli pozicionálás

(Folytatás a 26. oldalon)

Köszöntő

**Köszöntünk kedves Olvasó!
Ha végigrágod magad
ezen a néhány unalmas soron,
érdekes programokat, cikkeket
olvashatsz.
Mi érdekel? Arról is írunk!
Gépi kód, hardver,
szoftver, matematika, fizika,
HT, ABC–80 stb.
Amiről érdemes tudni,
és amiről nem, arról is.
Szereted törni a fejed?
Kitűnő, friss,
ropogós rejtvényeink vannak.
Persze mi sem vagyunk
mindenhatók és mindent tudók.
Ha valaki tud valami jobbat,
mint mi, írja meg,
és ne fossza meg tőle
a többi olvasót!
Aki pedig képes volt arra,
hogy ezt végigolvassa,
az el fogja olvasni egész
rovatunkat.
Jól teszi!**

A Z80 programozása

A legtöbb hazai gép Z80-as processzorral van ellátva. Aki komolyabban foglalkozik a számítástechnikával, vagy játékprogramokat szeretne írni, vagy olyan problémát akar megoldani, amely nagyobb gyorsaságot igényel, annak nagyon fontos a processzor ismerete.

A Z80 működése, programozása kezdők számára is megérthető. Azok is elkezdhetik elsajátítását, akik a magas szintű nyelveket nem ismerik.

Szeretnénk megkönnyíteni a gépi programozás elkezdését, ezért sok alkalmazási példát fogunk bemutatni, főleg az iskolaszámítógépekre.

Minden gépnek van memóriája. A memória legkisebb egysége a bit, melynek két állása van: a 0 és az 1. Nyolc bit alkot egy bájtot. Ha a biteket úgy értelmezzük, mint egy kettes számrendszerben írt szám számjegyeit, akkor látjuk, hogy nyolc bit segítségével a legnagyobb létrejövő szám 255. Az egész memória ilyen egységekből, bájtokból áll. Minden memóriabájtnak van címe. A memória egyes bájtjaira ezekkel a címekkel lehet hivatkozni.

A processzornak a memórián kívül regiszterei is vannak. A regiszterek felépítése hasonlít a memória felépítésére. Egy regiszter valójában egy bájt. A regiszterekre azért van szükség, mert a processzor a regiszterek tartalmával képes műveleteket végezni. A processzor ezenkívül képes még a regiszterek és a memória tartalmának cseréjére. A memória tehát arra jó, hogy ott adatokat tároljunk. Ha ezekkel az adatokkal műveleteket akarunk végezni, akkor a memóriában tárolt adatokat regiszterekbe töltjük, a regiszterekben elvégezzük a műveleteket, majd az eredményt ismét eltárolhatjuk a memóriában a további felhasználásig.

Egy-két szó a tárolásról. Ha egy bájt négy bitjét tekintjük, akkor látjuk, hogy ezen a négy biten 16-féle számot lehet ábrázolni. Ebből adódik az a gyakorlat, hogy a számokat 16-os hexadecimális számként írják. Hiszen egy bájton két darab hexadecimális számjegy ábrázolható. A hexadecimális számjegyek a következők: 1 2 3 4 5 6 7 8 9 A B C D E F. A memória minden bájtjának megcímzéséhez négy hexadecimális számjegy szükséges, tehát egy bájt címe két másik bájt segítségével tárolható, vagy két db regiszterben lehet rá hivatkozni.

Ahhoz, hogy gépi kódú programot írjunk, jól kell ismerni a memória felosztását. A programot ugyanúgy a memóriába írjuk, mint az adatokat.

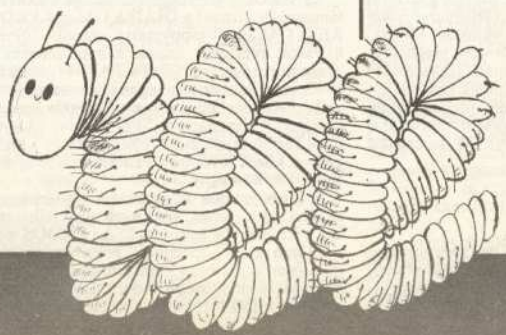
Térjünk rá a gépi programozásra. Említettük, hogy a processzornak regiszterei vannak. Ezek a következők: A, F, B, C, D, E, H, L, IX, IY, SP, PC, IR. A két betűvel jelzett regiszterek kettős regiszterek, vagyis kétbájtosak. Ezekre csak kétbájtos formában lehet hivatkozni. Az egy betűvel jelzett regisztereket külön is, de páronként, kettősregiszterként is lehet használni. Létezik tehát a HL, DE, BC, AF kettősregiszter. A PC programszámláló kettősregiszter. Tartalma mindig a következő végrehajtandó utasításnak a címe, tehát mindig az az utasítás hajtódik végre, amelynek címét a PC tartalmazza.

Az F regiszter regiszter-flageket tartalmaz, amelyek minden művelet után beállítódnak, aszerint, hogy mi a művelet eredménye. A flagek egy biten helyezkednek el. A következő flagek vannak: cary, zero, paritás, signum. Ezenkívül van még két flag, jelük N és H, ezeket azonban a processzor maga használja fel. A számunkra felhasználható flagek jele sorrendben: C vagy CY, Z, P, S.

A flagek a következőképpen állítódnak be. C akkor tartalmaz a neki megfelelő bit 1-et, ha a művelet során átvitel történik; különben 0-t tartalmaz. Ilyen eset például, ha kisebb számból nagyobbat vonunk ki. Ekkor a C flag tartalma 1. Z akkor 1, ha az eredmény 0. S akkor 1, ha az eredmény 0 vagy negatív.

Az így beállt flageket felhasználhatjuk úgy, hogy aszerint folytatódjon a program futása különböző helyeken, hogy az egyes flageknek mi az értékük.

Az IX és IY kettősregiszter indexregiszter. Értékük egy báziscím, amelyhez képest relatívan lehet megcímezni a memória egyes bájtjait.



lábú

A SP stack pointer segítségével zsákmemória készíthető. A SP a memória egy bájtjára mutat. Erre a bájtra tölthetjük be regiszterek tartalmát, vagy olvashatunk róla. Minden betöltés eggyel változtatja a SP értékét és minden kiolvasás ellentétesen változtatja a SP értékét. Tehát mindig azt az adatot olvashatjuk ki, amit legkésőbb tettünk bele.

Az IR regiszterek a megszakításnál használatosak. Ezekről később lesz szó.

Az A, B, C, D, E, H, L regiszterekben folyik maga a munka. Az A regiszternek kitüntetett szerepe van, neve akkumulátor. Ugyanilyen kitüntetett szerepe van a kettősregiszterek közül a HL-nek.

Az A, F, B, C, D, E, H, L regisztereknek vannak párjaik, amelyeket vesszővel jelölünk. Ezeket a vesszősregisztereket csak arra lehet használni, hogy a regiszterek és a vesszősregiszterek tartalmát kicseréljük. Műveleteket a vesszősregiszterekkel nem lehet végezni.

SIEBEN NÁNDOR

Polinom-illesztés

Valamelyik vasárnap este néztem a televízió Kapcsoltam című műsorát. A telefonrejtvény a következő sor két elemmel való folytatása volt: 1, 1, 1, 2, 1. (Lehet, hogy nem pont ez a számsor volt az, de valami hasonló.) Ha lenne egy számítógépem! – gondoltam, de nem volt.

Kedves olvasó, most biztosan azon töröd a fejed, hogy mire lett volna jó nekem a számítógép. Hát számítógéppel mindent meg lehet oldani? No nem! A számítógép betört az életünkbe, de mindent nem old meg, ellenben minden ilyen számsort lehet logikusan folytatni.

$$P(x) = \sum_{i=1}^{n} \frac{\prod_{i=1}^{j-1}(x-x_i)\prod_{i=j+1}^{k}(x-x_i)}{\prod_{i=1}^{j-1}(x_j-x_i)\prod_{i=j+1}^{n}(x_j-x_i)} y_i$$

```
10 REM POLINOM ILLESZTES
20 PRINT CHR$(12);"N=";
40 INPUT N
50 FOR I=0 TO N-1
60 PRINT "X(";I+1;")=";
70 INPUT X(I)
80 PRINT "Y(";I+1;")=";
90 INPUT Y(I)
100 NEXT I
110 FOR I=0 TO N-1
120 S=1
130 FOR J=0 TO N-1
140 A(I,J)=S
150 S=S*X(I)
160 NEXT I
170 A(I,N)=Y(I)
180 NEXT I
190 GOSUB 1000
200 FOR I=0 TO N-1
210 PRINT I;" FOKU EGYUTTHATO=";A(I,N)
220 NEXT I
230 STOP
1000 FOR I=0 TO N-1
1010 FOR J=I+1 TO N-1
1015 IF A(I,I)=0 THEN 2000
1020 S=A(J,I)/A(I,I)
1030 FOR H=I TO N
1040 A(J,H)=A(J,H)-S*A(I,H)
1050 NEXT H: NEXT J: NEXT I
1060 FOR I=N-1 TO 0 STEP -1
1070 FOR J=I-1 TO 0 STEP -1
1080 A(J,N)=A(J,N)-A(I,N)*A(J,I)/A(I,I)
1100 NEXT J
1110 A(I,N)=A(I,N)/A(I,I)
1120 NEXT I
1130 RETURN
2000 Z=I
2010 Z=Z+1
2020 IF Z>N-1 THEN 2080
2030 IF A(Z,I)=0 THEN 2010
2040 FOR S=0 TO N
2050 A(I,S)=A(I,S)+A(Z,S)
2060 NEXT S
2070 GOTO 1020
2080 END
```

A program az előbbi Lagrange-féle interpolációs formula alapján készült.

VERHÁS PÉTER

Szépítés az ABC–80-on

A következő szubrutin zenei hangokat generál. Két bemenő adatra van szükségünk: a hangmagasságra, M9% és a hang hosszúságára. A hangmagasság egy oktávra terjed ki, C-től kezdődően az értékei félhangonként: 45, 42, 40, 37, 34, 31, 29, 27, 25, 23, 21, 20, 18. A hosszúság értékei 1 és 16 közé eshetnek, a határokat is beleértve.

A szubrutin:

```
4000FOR L9%=1% TO 120%/SQR M9%
    xH9%
4010 OUT 6, 121
4020 FOR B9%=1% TO M9% : NEXT
    B9%
4030 OUT 6,0
4040 NEXT L9%
```

MARSI ANDRÁS

Program-csenés ZX81-gyel

Sokszor megtörténik, hogy olyan programot akarunk felvenni, amely betöltés után automatikusan indul. Ezt egyszerűen megtehetjük úgy, hogy belelépünk a LOAD ROM-rutinjába: FAST majd RAND USR 836. Ezzel rendesen beveszi a programot, de nem indul el, hanem miután bevette (vagy nem), C hibakóddal megáll. Ezután csak meg kell keresni a programban a SAVE utasítást, és GOTO-val odaküldeni.

BÉRES ISTVÁN

Rejtvény

A feladat egy értelmes magyar szó kitalálása az ASCII kód felhasználásával. A szó nagybetűkből áll. A feladatban szereplő összes szám hexadecimális.

Az első és a második betű különbsége 3.

A harmadik és az első betű különbsége A.

A harmadik és a második betű különbsége D.

A negyedik betű 4E.

A szó második felében a negyedik betűtől 1-gyel, illetve B-vel különböző betű található.

A szót a 3F-nek megfelelő jel zárja le.

(Folytatás a 23. oldalról)

végezhető karakterenként vagy mezőnként, előre és hátra irányban, mezőnkénti törléssel is. A bevitel ellenőrzött, hiba esetén jelzést kap a felhasználó. További sajátosság, hogy amíg a felhasználó nem lépett ki a megjelenített rekordból, addig mindig visszaállítható annak korábbi állapota.

Az UPDT hatékonyan támogatja outputok készítését is. Módosításnál az eredeti és módosított képek vihetők ki a nyomtatóra és/vagy új lemezes állományba. Minden más megjelenítésnél a billentyűzet lenyomásával nyomtatható ki a képernyő tartalma.

A hajlékonylemezes állományok mindig a kijelölt kulcsnak megfelelően rendezettek. Speciális algoritmus biztosítja, hogy a rekordok elérése igen hatékony legyen. Ez az egyik oka annak, hogy az állomány karbantartásának befejezésekor külön művelettel azt le kell zárni, minek következtében az állomány új állapotban rögzítődik a lemezen.

A TABL programcsomag az adatállományokból végzett tablókészítés általános eszköze. Nem rendezett állományok esetén az állomány az FMN06/08-cal rendezhető a tablókészítés előtt, külön menetben.

A TABL leválogatási funkciója az igények szerinti karaktersorozatoknak megfelelően működik. A karaktersor-részek számossága és elhelyezkedése a rekordon belül tetszőleges. A leválogatás szempontjai között tól–ig értéktartomány és a karaktersorozatok közötti ÉS, VAGY kapcsolatok szerepelhetnek. Egy karaktersorozat többször is szerepelhet a kapcsolatban. A leválogatott állományból előjeles összesen képzéssel egyszerű gyűjtés végezhető kívánság szerinti karaktersorokra.

A leválogatott állományra több karaktersorozat közötti ÉS kapcsolat adható meg összevont kulcs képzése céljából. Erre az új kulcsra tól–ig kijelöléssel válogatási szempont definiálható és erre gyűjtés (előjeles összegzés) végezhető. A leválogatott állomány szerkezete is (képernyőn, nyomtatón vagy hajlékonylemezes állományban) külön megadható. Fejlécet készíthető a leválogatott állományra, összegzésekre és gyűjtésekre is.

A TABL további szolgáltatása, hogy dátumkülönbség vizsgálható a futtatáskor megadott dátumhoz képest (például teljesítési elmaradás figyelése), a kulcs szerint gyűjtött értékekre azok összege szerint százalék képezhető, és a kiviteli mezőkből szorzat képezhető.

Az UPDT és a TABL használata úgy történik, hogy az adott alkalmazást funkcionálisan leíró jellemzőket a rendszerszervező táblázatos formában leírja (ún. deszkriptorok), majd egy assembly adatleíró direktívákat tartalmazó félkész forrásfájlba viszi be az EXDOS szövegszerkesztőjének segítségével. Az így kapott assembly adatmodult lefordítja, majd a kapott deszkriptor tárgykódot a Link segítségével összeszerkeszti a programcsomagok támogató eljárásainak tárgykódkészletével. A kész alkalmazás ezután szabadon futtatható.

A hazai fejlesztésű szoftvertermékek között igen figyelemreméltó az UPDT/TABL. Igen jó közelítését adja ugyanis egy általános alkalmazás generátor rendszernek. A TABL a generátor report writer részének tekinthető, az UPDT pedig a query/update nyelvnek. Rendelkezik továbbá a mikroszámítógépes alkalmazás generátorok azon tulajdonságával is, hogy magas színvonalon valósítja meg az előállítható alkalmazások párbeszédes használatát (rugalmas képernyő-kommunikáció). A felhasználó állományait a képernyőn keresztül látja, és azokban csak olyan módosítások történhetnek, amelyeket ő is lát (what you see is what you get). Az alkalmazásfejlesztés így nagyságrenddel hatékonyabb (csak szervezői munka), és a kész alkalmazás is a leginkább emberközeli módon használható.

Az alkalmazásfejlesztést közvetlenül támogató harmadik programcsomag BSC 2780 kompatibilis, transzparens adatátvitelt támogat és távoli munkabevitelt (remote job entry) tesz lehetővé nagyszámítógép felé. A kapcsolat eredményeképpen a Floppymat például Siemens, IBM vagy ESZR számítógép intelligens termináljaként képes működni. Rendelkezésre áll még az alapmódú szinkron, illetve aszinkron, transzparens átvitelt biztosító programcsomag is.

Az alkalmazásfejlesztési támogatás a jelenlegi hazai szűkös ellátottsági viszonyok között igen jó helyzetet biztosít a Floppymatnak. Az UPDT/TABL révén olyan alkalmazásgenerátora is van, amely – tudomásom szerint – jelenleg egyetlen hazai mikroszámítógépen sem áll rendelkezésre. A nyugati piacon kapható alkalmazásgenerátorok természetesen jóval fejlettebbek és érettebb a konstrukciójuk. Adatbázis-koncepcióra épülnek, így jóval általánosabbak. Ez az UPDT/TABL-ról nem mondható el. A deszkriptorok megadása is történhetne párbeszédes módon, a képernyőn keresztül, ami az assembleren és linkeren keresztüli használatot is feleslegessé tenné. Ezek azonban mind olyan észrevételek, amelyek – hazai példa nem lévén – nem róhatók fel az UPDT/TABL-nak.

Alkalmazási rendszerek

A Floppymatokon jelenleg csak hagyományos jellegű alkalmazási rendszerek állnak rendelkezésre. Ezeket a rendszereket a gyártóval szorosan együttműködő szervezési és számítástechnikai szervezetek fejlesztették ki egy-egy célalkalmazásra.

A VSZFT tanácsi ügyirat-nyilvántartó és iktató rendszert fejlesztett ki, amely egy sor budapesti kerületi tanácsnál és egyes vidéki tanácsoknál most áll bevezetés alatt, vagy már be is vezették. Kifejlesztették és részben már be is vezették az anyagkészlet-nyilvántartást és elszámolást, vevői rendelés-nyilvántartást és szállítói rendelés-nyilvántartást végző rendszert is. Tanácsok részére személyi nyilvántartási rendszert dolgoztak ki.

A SZÖVORG bolti elszámoló rendszert, a MÜSZI a mezőgazdaság területén alkalmazható rendszereket (munkaügy, pénzügyvitel, szállítók nyilvántartása) fejlesztett. Az OKISZ SZSZV a BKV részére bérszámfejtő típusrendszer kidolgozásával foglalkozik. A PM SZÜV főkönyv vezetésére, folyószámla kezelésére és utókalkuláció végzésére készített rendszereket (jelenleg VILATI alkalmazás). Alkalmazási rendszerek fejlesztésével foglalkozik még a DATORG is.

A fenti rendszerek túlnyomó része típusrendszerként készül, azaz megfelelő adaptációs munkával igen sok vállalatnál bevezethetők. A rendszerek egy része már most is kapható, és mire ismertetőnk megjelenik, már nagyobb részük megvásárolható lesz.

Az alkalmazási rendszerek választékát elég széles körűnek ítélhetjük. Néhány hiányzó elem: szófeldolgozó rendszer (word processor) és elektronikus feladatlap (electronic spreadsheet).

Kereskedelmi és vevőszolgálati kérdések

Az egyes berendezések ára:

Floppymat–E	295 eFt
Floppymat–I	390 eFt
Floppymat–SP	460 eFt
iker meghajtós bővítés	182 eFt
DZM–180 nyomtató	195 eFt
SZM 5400 kazettás lemez	497 eFt
további egységek (max. 3)	387 eFt
Floppymat Konverter	985 eFt
PK–1 digitális kazetta	160 eFt.

A szoftver rendszerek ára:

FMN06/08 a szállított konfiguráció árában	
EXDOS valamennyi eszköze a szállított konfiguráció árában	
Ügyviteli Pascal rendszer futtató rendszer	20 eFt
teljes rendszer	100 eFt

UPDT/TABL betanítással:

Rendszerfejlesztő vállalat részére	120–250 eFt
Egyedi felhasználó részére	50–70 eFt
Adatátviteli csomag (algoritmusonként)	65 eFt

Az alkalmazási rendszerek árai változóak, és a terjesztő vállalat határozza meg őket az adaptálási munka mértékétől függően.

A Floppymatok szervizével részben maga a gyártó, alternatív lehetőségként pedig a Fejér megyei Műszerész Ipari Szövetkezet foglalkozik. A köthető szerződés általában 2 napon belüli megjelenést és 6 napon belüli javítást vállal. Szükség esetén ennél szigorúbb feltételek mellett is köthető karbantartási szerződés.

Kereskedelmi és vevőszolgálati tekintetben a Floppymatok kb. azt a színvonalat biztosítják, amit a jobb hazai szállítók vállalnak. Az árfekvés azonban a Floppymatok esetében is elég magas; kívánatos lenne ennek csökkentése.

Végső következtetések

A Floppymat gépcsalád megfelelő hardver és szoftver támogatással rendelkezik ahhoz, hogy a helyi feldolgozási igényeket irodai/ügyviteli számítógéppel ki tudja elégíteni. A VILATI széles körű együttműködésre is törekszik más rendszerfejlesztő vállalatokkal (Floppymat Fejlesztési Társulás és más kooperációk), ami a továbbiakban is biztosítani fogja a szoftverválaszték bővítését.

A jelenlegi Floppymat-konstrukció azonban középtávon problémákat okozhat mind a gyártónak, mind a felhasználóknak. Minél inkább elterjednek ugyanis hazánkban a hardver/szoftver rendszerükben nyitott és szabványos konstrukciójú rendszerek (például 8080/Z80 bázisú CP/M), annál inkább elkülönül a Floppymat. Ez csökkenti a potenciálisan megvásárolható szoftverválasztékot, növeli az alkalmazások árát, és önmagában is kiváltója lehet a szabványos rendszerekre való költséges átállásnak.

NACSA SÁNDOR

Mikroszámítógépek a számviteli-pénzügyi adatfeldolgozásban

A PM Szervezési és Ügyvitelgépesítési Vállalat – ügyvitelszervezési és gépesítési tevékenységen belül hangsúlyozottan – adatfeldolgozásra alkalmas mikroszámítógépekre kidolgozott *alkalmazási programcsomagokkal* áll megrendelői szolgálatára a számviteli és a pénzügyi munka területén. A mikroszámítógépek és a programcsomagok teljesítménye és ára igazodik a megrendelő által kitűzött feladatokhoz, valamint anyagi lehetőségeihez. Amennyiben a feladat megköveteli, a programcsomagok nyilvántartási kartonokat kezelnek, ezt az adatfeldolgozásra alkalmas mikroszámítógépek kartonelőtét-berendezése teszi lehetővé. A programcsomagok használatba vétele kb. 25%-ára csökkenti az adatfeldolgozás gépesítéséhez a felkészülési időt, valamint a szervezési és programozási munkát, 20%-ára csökkenti a gépesítés bevezetésének költségeit. A programcsomagok az államigazgatási, ezen belül elsősorban a pénzügyminisztériumi szabályozások figyelembevételével készültek és a gazdaságirányítás módosításai alapján folyamatosan aktualizálásra kerülnek. A rendelkezésre álló programcsomagok és a működtetést biztosító gépek:

Főkönyvi és folyószámla-könyvelés
ROBOTRON A 5110, A 5130, 1750, 1720, 1720 MKA, 1720 FD, 1711/16, 1711/64
FELIX FC–64, FC–96/128

Mérlegbeszámolóhoz adatszolgáltatás
ROBOTRON A 5110, A 5130

Analitikus anyagkönyvelés
ROBOTRON A 5110, A 5130, 1750, 1720, 1720 MKA, 1720 FD, 1711/16, 1711/64
FELIX FC–64, FC–96/128

Készletnyilvántartások mezőgazdasági vállalatok részére
ROBOTRON 1720

Fogyóeszköz-nyilvántartás
ROBOTRON A 5110, 1750, 1720 MKA, 1720 FD, 1711/64

Késztermék- és félkésztermék-könyvelés
ROBOTRON 1711/64

Késztermék-nyilvántartás és számlázás
ROBOTRON 1750

Késztermék-nyilvántartás és számlázás mezőgazdasági vállalatok részére
ROBOTRON 1750

Bérelszámolás
ROBOTRON A 5110, 1750, 1720 MKA, 1711/64,
FELIX FC–64, FC–96/128

Állóeszköz-könyvelés
ROBOTRON A 5110, A 5130, 1750, 1720, 1720 FD, 1711/64
FELIX FC–64, FC–96/128

Boltelszámolás (szövetkezeti)
ROBOTRON A 5110, 1720

Háztáji gazdaságok elszámolása (mezőgazdaság)
ROBOTRON 1711/64

Számlázás
ROBOTRON 1720, 1720 FD, 1711/16, 1711/64
FELIX FC–64, FC–96/128

Számlázás (zöldség, gyümölcs kereskedelem)
ROBOTRON 1720 FD

Rendelésnyilvántartás, számlázás
ROBOTRON A 5110, A 5130

Költségvetési szervek előirányzatai és elszámolásai
ROBOTRON 1750, 1720

Illetéknyilvántartás és elszámolás
ROBOTRON 1720

Forgalmiadó-elszámolás
ROBOTRON 1720

Gyártáselőkészítés (gépipari)
ROBOTRON 1750

Szövegfeldolgozás
ROBOTRON A 5110, A 5130

PM Szervezési és Ügyvitelgépesítési Vállalat 1052 Budapest Tanács krt. 20.

Bemutató terem címe: Budapest VI., Paulay Ede u. 15.

Kirendeltségek: 3530 Miskolc, Vándor Sándor u. 2.
9021 Győr, Tanácsköztársaság u. 15.
7400 Kaposvár, Tóth Lajos u. 12.
5600 Békéscsaba, Tanácsköztársaság u. 24.

Szövetkezeti Gazdaságszervezési és Számítástechnikai Iroda

1253 Budapest 13. Postafiók 56.

Központ: 1013 Budapest I., Attila út 13.
Telefon: 189-271, 189-162
Telex: 22-4263
Igazgató: 189-240
Igazgatóhelyettes: 189-404
ÜZEM- ÉS MUNKASZERVEZÉSI FŐOSZTÁLY:
Bp. V., Honvéd u. 16. T.: 113-017
REKLÁMSZERVEZÉSI OSZTÁLY:
Bp. VIII., Tömő u. 48-54. T: 340-771
NYOMDA:
Bp. VIII., Bacsó Béla u. 39. T: 342-953

Alábbi tájékoztatónkban nagyon rövid ismertetést adunk Irodánk szervezeti egységeinek szolgáltatásairól, azzal a céllal, hogy megkönnyítsük az Önök számára az igényüknek megfelelő szakosztályunkkal való kapcsolatfelvételt. Irodánk megrendelésre dolgozik, tevékenységünk kialakításánál rugalmasan igazodunk a megbízóink által támasztott követelményekhez. Nagy súlyt helyezünk arra, hogy szervezési javaslataink, szolgáltatásaink színvonala megfeleljen a legkorszerűbb hazai és nemzetközi gyakorlatnak. Megtiszteltetésnek vesszük, ha a problémái megoldásához igénybe venni szándékozott külső szakértők között bennünket is megkeres ajánlatkérésével. A profilunkba tartozó bármilyen új, korszerű téma kidolgozásához szívesen csatlakozunk saját költségünkre, vagy alacsony díjtérítés mellett is. Kívánságra munkánkról referenciahelyet biztosítunk.

ÜGYVITELSZERVEZÉSI ÉS SZÁMÍTÁSTECHNIKAI FŐOSZTÁLY

A döntő részarányt képviselő ÁFÉSZ és szövetkezeti vállalati megrendelőinken túl, más ágazatokhoz tartozó gazdálkodó szervek részére is nyújtjuk az alább felsorolt szolgáltatásainkat. A főosztály feladatra orientált team-rendszerben dolgozik. Alapadminisztráció, könyvelési és ügyviteli folyamatok, valamint feldolgozási módok folyamat-, rendszerszervezése, programozása, alkalmazói és számítástechnikai feladatok komplex kivitelezése.

Egységes mintarendszerek kialakítása és dokumentálása, adaptálása és koordinálása. A könyvelési, feldolgozási, a komplex információs rendszerhez kapcsolódó ügyviteli, alkalmazási és számítástechnikai eszközök beszerzése, értékesítése, kölcsönzése, ezzel kapcsolatos alkalmazási feladatok koordinálása. Szabványos és nem szabványos, gépi és kézi bizonylatok és nyomtatványok gondozása, gyártása és forgalmazása. Gazdálkodó egységek kód- és számrendszerének, alapadminisztrációjának egységes kialakítása, gondozása és fejlesztése. Szervezéshez kapcsolódó pénztárgépek és kellékek beszerzése, forgalmazása, oktatás és üzembe helyzés koordinálása. Gépkezelők, operátorok, szervezők és programozók, valamint egyéb alkalmazói és számítástechnikai szakemberek képzése, továbbképzése és speciális képzése. Alkalmazástechnikai szolgáltatás szervezése, gyakorlati bemutatóterem biztosítása, alkalmazástechnikai rendszerek, módszerek, dokumentációk és programcsomagok témára és gépre orientált bemutatása. Saját gépparkunk és oktatótermünk díjtérítés ellenében rendelkezésükre áll. Gépeink: VT 20/A, Floppymat SP, ROBOTRON 5130, 5110, 1370–72, 1711/256, 1111/64, 1720+FD+LS, 1355, ASCOTA 170/35, Commodore 64.

KÖZGAZDASÁGI FŐOSZTÁLY

Üzemgazdasági Osztály

A szövetkezetek és vállalatok belső irányítási rendszerének és működésének felülvizsgálata, korszerűsítése, áruforgalmi, hálózatirányítási, felvásárlási, tervezési, személyügyi, elszámoltatási, pénzügyi és ellenőrzési folyamatának szervezése. Szervezeti és működési szabályzat, valamint egyéb belső szabályzatok, munkaköri leírások elkészítése. Ösztönző és hatékony érdekeltségi rendszerek kidolgozása. Gazdálkodó szervek egész tevékenységét átfogó komplex szervezési munkák vállalása. Új vállalkozási formák bármelyikének szervezése, jogi-közgazdasági megalapozása, eljárási forgatókönyv biztosítása.

Tanácsadó Szolgálat

Csoportos konzultációk szervezése éves előfizetéses rendszerrel, melyek témái időszerű szakmai, számviteli, adóügyi, pénzügyi, statisztikai, munkaügyi kérdéseket, rendeletértelmezést dolgoznak fel. A konzultációkhoz tapasztalatcseréket is szervezünk. A csoportos konzultációkra előfizető megrendelőinknek gyakorlati tudnivalókat tartalmazó szakanyagokat, módszertani útmutatókat, oktatási segédleteket adunk ki. ÁFÉSZ elnökök Fórumának szervezése.

Kiadói és Információs Osztály

Szakmai tanulmányok kiadása, aktuális vezetési, szervezési és gazdálkodási ismeretek, módszerek tárgyában. Az érvényes jogszabályok egységes szerkezetbe foglalt szövegének megjelentetése, kiegészítve a jogszabályok értelmezésével. Mintaszabályzatok kiadása. Az élenjáró szervezési módszerek dokumentálása és megjelentetése. A nemzetközi szakirodalom hasznosítható tapasztalatainak esetenkénti kiadása. Hazai és külföldi szakirodalom figyelése, ezek esetleges tematikus feldolgozása.

Munka- és Üzemszervezési Főosztály

Kiskereskedelmi hálózati egységek korszerűsítését célzó technológiai tervdokumentációk összeállítása, ezek kivitelezésének szakmai segítése. Hálózatfejlesztési igények közgazdasági számításokkal való megalapozása, beruházási tevékenység technológiai tervezési feladatainak ellátása. Kereskedelmi hálózati egységek közötti munkamegosztás fejlesztése, az üzleti profilok meghatározása. Háttéripari, ipari szolgáltatási tevékenység beindításával, hatékonyságának javításával összefüggő szervezési feladatok ellátása. A mezőgazdasági termékek termeltetésével és forgalmazásával összefüggő szervezési feladatok segítése.

SZOLGÁLTATÁSI FŐOSZTÁLY

Vevőszolgálati Osztály

A SZÖVORG gazdasági partnereivel kapcsolattartás, más gazdálkodó szervezetekkel közösen vállalt feladatok szervezése. A SZÖVORG szolgáltatásaival kapcsolatos panaszok összegyűjtése, szintetizálása, a megoldás érdekében teendő intézkedések meghatározása.

Reklámszervezési és Technikai Osztály

Reklám-propaganda tevékenység tervezése. Hirdetések szervezése. Cégfeliratok, tájékoztató táblák. Dekorációs eszközök igény szerinti egyedi gyártása és felszerelése. Csomagolástechnikai eszközök termeltetése és forgalmazása. Propagandaanyagok, ajándéktárgyak forgalmazása. Kiállítások, árubemutatók szervezése, kivitelezése.

Nyomdaüzem

Nyomtatványok ○ Beszámolók, szabályzatok, előterjesztések ○ Árupropaganda célokat szolgáló prospektusok ○ Árukiszerelő címkék ○ Meghívók ○ és egyéb nyomdatermékek előállítása.

Terminálok

A FELHASZNÁLÓ SZEMSZÖGÉBŐL

Milyen legyen az ideális terminál? – ezzel a kérdéssel fekszenek és kelnek a termináltervezők. Hát legyen rajta sok gomb, legyen kényelmes a kezelése. Tudjon jól „szöveget szerkeszteni", hiszen a programozás tulajdonképpen egy többé-kevésbé kötött textus előállítása. Aztán jó, ha lehet törölni, betoldani, cserélni. Nem árt, ha a cursor minden irányban könnyen mozgatható. És a képernyő! Lehetőleg az alsó sorban (vagy külön lámpákon) sok információt adjon „lelkiállapotáról", hogy most éppen mit is gondol a világról. Súly, méret, csatlakozók? Nem nagyon lényeges, az ember majd keres neki helyet az irodában. Legfeljebb az fontos, hogy ne legyenek a tetején vízszintes felületek, mert akkor rákerül a leporelló meg a kávés pohár, és ez nem mindig előnyös.

Valóban így születik egy terminálcsalád? Valószínűleg nem, de az eredmény rendkívül egyezik a fenti gondolatmenetből adódóval. Ez talán nem is volt tragikus – egészen máig.

Miért más a „ma terminálja"? Miért sántít a fenti logika? Nos, egyetlen okból: hazánkban is elkezdődött nagyobb arányban (végre!) az on-line rendszerek telepítése, és így a terminál alapvetően új környezetbe került. Az irodai környezet még csak-csak megmaradt (bár a műhelyterminálok problémáit tárgyalhatnánk!), de a kezelők! – Hát igen, le kell szoknunk arról, hogy a kezelők a számítástechnika „professzorai". Ők egyéb témák szakértői: biztosítási ügyintézők, bankalkalmazottak, gazdasági osztályok munkatársai, vagy tervezőmérnökök, akik a szakterületüket kiválóan ismerik, készségük és hajlamuk van az új technika elsajátítására, de a számítástechnikát csak eszköznek tekintik feladataik elvégzéséhez.

Nekünk – ha ismert és ismeretlen kollégáim megengedik nekem ezt a többes számot – a számítástechnika önmagában szép; talán akkor is szívesen foglalkoznánk vele, ha senkinek sem lenne rá szüksége. De van! Ezért úgy kell végeznünk a dolgunkat, hogy a felhasználó, az „end-user" jól érezze magát.

Ebben a felhasználóért folyó harcban egyik döntő fegyverünk a terminál. A legeslegjobban tervezett on-line rendszert is képes tönkretenni egy ergonómiailag elrontott, gyakran meghibásodó és felesleges funkciókkal telezsúfolt végkészülék.

Azt hiszem, a többi kívánság (ergonómia stb.) sokszor letárgyalt téma, de keveset foglalkozunk a célterminál kérdésével. A felhasználó nem akkor boldog, ha terminálja tele van számára szükségtelen, zavaró nyomógombbal; olyanokkal, amelyeknek megnyomása még bajt is okozhat (például dumpol a program vagy leáll a vonal stb.).

Legyen a terminál áttekinthető felépítésű, mentes minden felesleges „karácsonyfadísztől". Ezek a „díszek" fontosak a programfejlesztőknek, de nem az ügyintézőknek, a számításaira koncentráló mérnöknek, a kérdésére választ váró kutatónak.

Azt hiszem, a fentiekből adódik, hogy az általam ideálisnak tekintett termináltípus rendkívül komfortos, tud „mindent", de a gyártó álljon készen arra, hogy ezen a készleten belül a legkülönbözőbb igényeket gyorsan ki tudja elégíteni.

Milyenek lehetnek ezek az igények? Bemutatok néhány konkrét példát.

Egyik on-line programunk olyan, hogy a következő kép kiadását csillag beküldésével kell kérni. Az általunk alkalmazott VDT terminálon (a beszerzés idején ez volt kapható) a csillag a nagybetűs sorban volt, vagyis SHIFT-tel együtt használható; alul ezen a gombon a kettőspont található, amire esetünkben semmi szükség nem volt. Apró kérés a terminálon dolgozó munkatársak részéről: cseréljük meg ezt a két karaktert. Átkainkat mindenki ismeri, aki foglalkozott már idegen mikroprogramok viszszafejtésével és módosításával. A kérés azonban értelmes és jogos volt: a két betű helyet cserélt. (Szerencsére mind a két jel szimmetrikus, így a gombsapkát elegendő volt megfordítani.)

Egy másik esetben az insert és delete funkciókat kellett kiirtani, hogy ne zavarhassák a program működését. (Hogy ki lehet az ilyet programmal is védeni? Valóban, de ez a megoldás éppen az on-line esetben kritikus válaszidőt növeli.)

Ilyen és hasonló kérések minden rendszer elkészítésénél felmerülnek. Deklarálni kellene, hogy ezek az igények nem a felhasználó agylágyulásának korai jelei; ezek az igények jogosak, és már a rendszer fejlesztésének kezdeti szakaszában figyelembe veendők.

És akkor még nem is beszéltünk az egy-egy konkrét rendszer által igényelhető funkcionális nyomógombok szerepéről. Pedig ilyen „jelentéktelen" szempontok érvényesítésétől, a gyakorlatias megoldásoktól lehet az egyik projekt siker, a másik kudarc – és kudarcunk már annyi volt!

Ideje lenne végre jó rendszereket készíteni, amihez sok sikert kíván az olvasónak (tervezőnek és felhasználónak egyaránt):

DR. NAGY ÁKOS

Karaktersorozat-keresés BASIC-ben

Szövegszerkesztésnél, szövegfeldolgozásnál gyakran előforduló feladat, hogy egy karaktercsoport előfordulási helyét meg kell keresni valamilyen szövegben. Ennek célja lehet a kigyűjtés, a módosítás stb.

Abban az esetben, ha csak azt kell megállapítanunk, hogy karaktersorozatunk mint teljes sorozat előfordul-e a sorozatkészletben, a feladat egyszerű. Össze kell csak hasonlíttatni a keresett és a létező sorozatokat.

Legyen A$ a keresett sorozat, B$ (I, J) a készlet egy létező sorozata, M és N a létező sorozatok tömbjének maximális indexe. Ebben az esetben a programrészlet például:

```
10 INPUT A$
20 FOR I=1 TO M: FOR J=1 TO N
30 IF B$ (I, J) < > A$ THEN 50
40 PRINT I, J
50 NEXT J: NEXT I
```

(Amennyiben az általunk használt BASIC-változatban sorozatok IF utasításon belül nem hasonlíthatók össze, úgy a 30., 40. sor a következő programrészlettel helyettesíthető:

```
30 U=0: V=LEN(A$): FOR K=1 TO V
35 IF ASC(MID$(B$(I, J), K, 1))=
   ASC (MID$(A$,K,1)) THEN 45
40 U=1:K=V
45 NEXT K: IF U=0 THEN PRINT I,J
```

ahol U, V segédváltozó, a LEN-függvény a sorozat hosszát adja meg, az ASC-függvény egy karakter ASCII-egyenértékét adja – tehát egy számot, ami már minden BASIC-változatban összehasonlítható –, a MID$-függvény a sorozatból a vesszők utáni paraméterek által megadott kezdő pozíciótól, megadott számú karakterből álló sorozatot választ ki.)

Nehezebben oldható meg az a feladat, amikor azt is meg akarjuk tudni, hogy a keresett sorozat, mint egy sorozat része előfordul-e a sorozatkészletben (például legyen a keresett sorozat „MA"; ez előfordulhat mint szó, szóvégi szótag – például „ALMA" –, szó eleji szótag – például „MAI" –, szó belsejében levő szótag – például „ALMAMAG" –, szótagrész – például „MANKO", ami nem sajtóhiba, hanem a BASIC-ben nincs ékezet – stb). A BASIC-változatoknak csak egy kis része rendelkezik az ilyen típusú keresést lehetővé tevő utasítással. Például a Commodore C–64 típusú géphez külön megvehető ún. Simon's BASIC, amelynek PLACE-utasítását használva eredeti programrészletünkben a 30., 40. sor helyett, a következő lesz:

```
30 U=PLACE(A$,B$(I,J)):
   IF U=0 THEN 50
40 PRINT I,J,U
```

Az U=Ø azt jelenti, hogy ez a sorozat nincs a vizsgálatban. Itt eredményként nemcsak azt kapjuk meg, hogy mely sorozatokban fordul elő a keresett, hanem azt is, hogy hanyadik karaktertől kezdődően.

Amennyiben ilyen utasítás az általunk használt változatban nincs, úgy eredeti programrészletünk 3Ø., 4Ø. sora helyére a következőt helyettesíthetjük be:

```
3Ø IF LEN(A$)>LEN(B$(I,J)) THEN 5Ø
35 FOR K=1 TO LEN(B$(I,J))-
   LEN(A$)+1
4Ø IF MID$(B$(I,J),K,LEN(A$))<>A$
   THEN 48
45 PRINT I,J,K
48 NEXT K
```

Ez a programrészlet először azt vizsgálja, hogy nem túl hosszú-e a keresett sorozat. Ha nem, úgy azzal azonos hosszúságú összefüggő sorozatokat választ ki a vizsgált sorozatból az elsőtől az utolsó még lehetségesig, és ezeket hasonlítja össze a keresettel. Kedvező esetben itt is megkapjuk a talált sorozat kezdő helyét is. (Amennyiben egy sorozaton belül többször is előfordul a keresett – mint például a „MAMA" sorozatban –, úgy több eredményt is kapunk.)

S. E.

ZX81-tulajdonosok, figyelem!

A BREAK/SPACE billentyűpárosítás gyakran okoz problémákat, ha az INKEY$ függvényt használjuk.

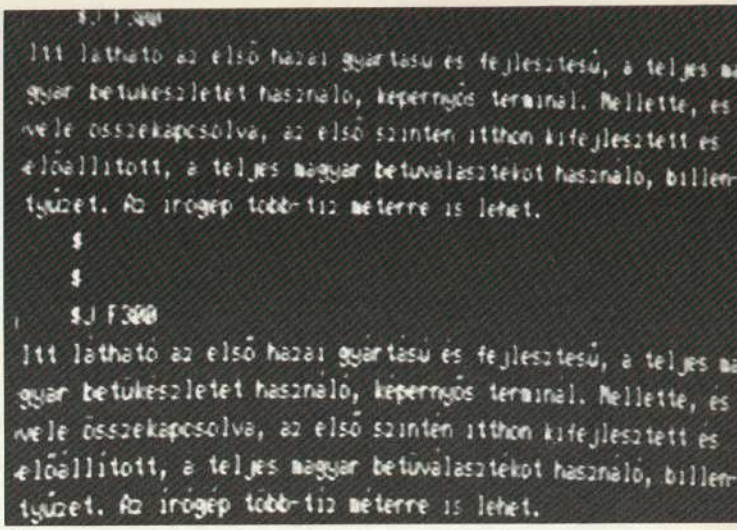
A megoldás egy program és az ENTER/NL billentyű használata a SPACE-funkció megvalósítására:

```
2Ø PAUSE 4ØØØØ
3Ø IF INKEY$=CHR$ 118 THEN
   PRINT "≠≠";
4Ø IF INKEY$=CHR$ 118 THEN
   PRINT INKEY$;
5Ø RETURN
```

A szubrutin 4Ø. sorában 118 helyett 64 is lehet. Ez a szubrutin kiküszöböli a nem kívánt ún. kulcsszó-kiírásokat (kivéve azt, hogy a SHIFT Q és SHIFT P között nincs különbség).

A SHIFT LOCK billentyű kényelmesebbé teszi a bebillentyűzést. Fúrjuk át a billentyűzet bal sarkát a SHIFT billentyű felett. Itt van elég hely egy kisméretű billentyű számára. Helyezzük az új billentyű megfelelő lábait a billentyűzet gumicsatlakozójára. Vezessünk át vezetéket az alsó oldalra. Két gumikábel vezet ki a billentyűzetből. A bal oldali öt, a jobb oldali nyolc vezetékből áll. Az egyik vezetéket a bal oldali csoport legjobboldalibb vezetékéhez, a másikat a jobb oldali csoport jobbról számított harmadik vezetékéhez forrasztjuk.

S. E.



Építsünk számítógépet! I.

Először is röviden felsoroljuk a számítógéprendszer egyes egységeit, és elsőként ismertetjük az ún. billentyűzet-illesztő egységet. Számítógépünk egy egyetlen áramköri kártyán megvalósított mikroszámítógépből, egy speciális billentyűzet-illesztőből és billentyűzetből, közönséges tv-vevőből és közönséges magnetofonból áll.

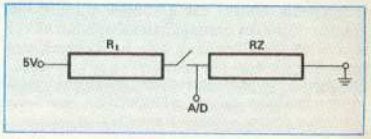
A legtöbb neves szaklap közölt hasonló tárgyú sorozatot, melynek első néhány részében elektronikai alapfogalmakat, egyes fontosabb áramköröket, majd a mikroszámítógépet ismertette. Ezután következett a gép működtetéséhez feltétlenül szükséges alapprogramok leírása, és végül a használathoz elengedhetetlen periféria-illesztők (billentyűzet-, kijelző-, háttértároló-illesztő) bemutatása.

Mi ettől eltérő sorrendű ismertetést adunk. Elsőként foglalkozunk a billentyűzet-illesztővel, mert: 1. egyszerű felépítése miatt könnyebb megérteni és megépíteni; 2. a számítógép nélkül is kipróbálható; 3. a majdani számítógép üzembe helyezését, kipróbálását megléte leegyszerűsíti; és talán ez a legfontosabb, 4. bármilyen számítógéphez használható, így azok számára is hasznos lehet, akik nem akarnak számítógépet építeni. (Figyelmeztetnünk kell az olvasókat arra, hogy a megoldás csak a szerző és feltalálótársai engedélyével hasznosítható!)

Az illesztő alkalmas arra, hogy *bármilyen mozgószáras billentyűjű billentyűzetet számítógép-billentyűzetté alakítson.* Így akármilyen (mechanikus is!) írógép, számológép, könyvelőgép, kártyalyukasztó stb. használható, természetesen csak akkor, ha műszaki állapota megfelelő.

A billentyű lenyomása egy vagy több áramkör valamely meghatározott pontján levő egyenfeszültségeket az áramkör zárásával vagy nyitásával megváltoztatja, és ezzel – az erre a pontra kötött – számítógép-párhuzamos bemeneten jelállapot-változást hoz létre. A változásokból alkalmas programmal megállapítható, hogy melyik billentyű került lenyomásra.

Az illesztőben az áramkörök körönként

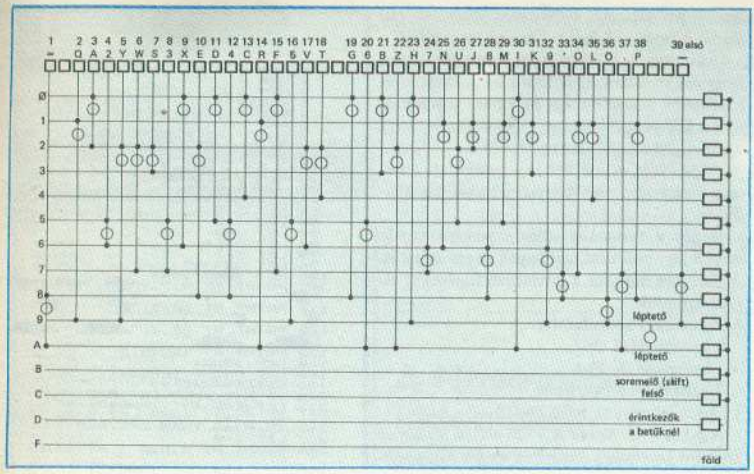


1. ábra

egyenfeszültségű áramforrást (például a számítógép +5 V tápfeszültsége és földje), egy ellenállásosztót, az osztáspontot a számítógép párhuzamos bemenetével összekötő vezetéket, az osztáspontnál levő záró-nyitó érintkezőt és ez utóbbit a billentyűvel összekötő részt tartalmaznak *(1. ábra)*.

A megoldás előnyei:

- viszonylag olcsó,
- a billentyűzet eredeti funkcióját (ha például írógép-átalakítás történt) nem zavarja,
- írógép használata esetén a bebillentyűzött adatok papíron is rögzítődnek,
- az egyenfeszültség használata miatt a megoldás sokkal kevésbé érzékeny a zavarásokra, mint mások, ezért a billentyűzetet a számítógéptől messzire is lehet telepíteni,



2. ábra

• az átalakításos megoldás előnye az új billentyűs egységekre történő cserével szemben (a kisebb költségen és a felsoroltakon felül) az, hogy a kezelőnek nem kell egy új berendezés kezelését megtanulnia.

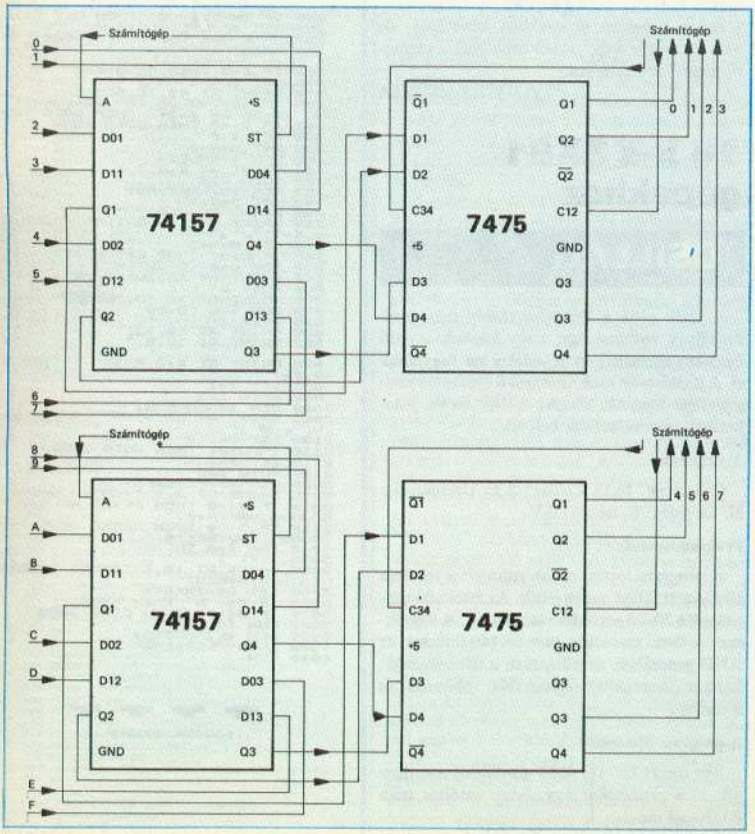
A megépített megoldásnál összesen 42 írógépbillentyűt (beleértve a soremelőt, léptetőt és egy speciális célokra használt billentyűt) használtunk. Ezek segítségével a szabványos 7 bites ASCII-kódok közül 120-at használhatunk. Ugyanez a megoldás alkalmas a teljes magyar betűkészlet alkalmazására is.

Az illesztő billentyűzetnél levő részének elvi rajzát a *2. ábra* mutatja. Az ábra felső részén látható a billentyűzet megnevezése, bal oldalon a vezetékek hexadecimális sorszáma. A karikák az egyes érintkezőket jelzik.

A számítógépnél található egyszerű áramkört a *3. ábrán* szemléltetjük. A 74157 ún. multiplexer, a 7475 ún. latch. A két egységet egy 15 erű kábel (célszerűen szalagkábel) köti össze.

Azoknak az olvasóknak, akik nem ismerik a cikksorozatban használt fogalmakat, a szerző szívesen nyújt tájékoztatást. Ugyancsak a szerzőnek írjanak azok (a Szerkesztőség címére) akik a számítógépet vagy valamelyik részegységet meg akarják építeni.

DR. SIMONYI ENDRE



3. ábra

Válasz Katona László levelére

Most is, és a továbbiakban is a levelek többeket érintő kérdéseivel, megjegyzéseivel kapcsolatos szerkesztőségi válaszaink a lap következő számában jelennek meg.

Katona László a lap első, 1983. évi számában megjelent két cikkel és néhány, a cikkektől független gondolattal keresett meg bennünket. A „VIC–20 memória kiterjesztése" című írásról kérdezi, hogy a 4. ábrán az IC-k bekötése párhuzamos-e? Az adat- és cimbusz szempontjából igen. Ugyanezzel az ábrával kapcsolatban megjegyzi, hogy a RAM2 és RAM3 jelű IC-nek 19 lába van. Ezt a megjegyzést köszönjük. Rajzolási hiba, hogy a V_{CC} pontra két vezeték (a tápfeszültség és a RAM1 címzővonal) van kötve. Az utóbbi téves.

A ZX81 adattárolásával foglalkozó cikkel kapcsolatos kérdése, hogy a programok 12Ø., 7Ø., 4Ø. sorainál mit kell beadni? Ezeken a helyeken a gép a kazettaegység kézi indítását várja, tehát akár számot, akár betűt beadhatunk az indítás után. Kérdezi, hogy a 17Ø. sorban hiányzik-e valami az = és a + jel között? Igen, lemaradt egy D betű. Véleménye szerint ezen a gépen nincs ENTER és DELETE billentyű. Egyes ZX81 sorozatoknál ezeken a billentyűkön NEWLINE, illetve RUBOUT felirat van.

A ZX81 (Spectrum) géppel kapcsolatos egyéb problémáival keresse meg a HCC Sinclair Szekciót (minden hétfőn fél 6 és fél 8 között a Kertészeti Egyetemen, Bp. VI., Eötvös u. 51.).

A VIC–20 (és más Commodore gyártmányú gépek, például C–64, PET, CBM) az újonnan megalakult Commodore Használók Szekciója egyik fő témája lesz. A szekció találkozóinak helye és időpontja felől az Egyesület titkárságán lehet érdeklődni.

Újonnan alakult szekció még a PTA–4000/PC 1500 Használók Szekciója is.

DR. SIMONYI ENDRE

Köszönjük Olvasóink leveleit, az ötleteket és a programokat. Kérjük egyúttal, hogy aki közlésre szánt hosszabb programot küld be, mellékeljen egy kazettát is, melyet természetesen visszaküldünk. A könnyebb olvashatóság kedvéért bátran használják a REM-et is!

1 k-s ZX81 gépekhez

SZÓPÓKER

Nem akarok a szükségből érdemet kovácsolni, amikor azt állítom, hogy igen tanulságos a készséges, ám rövid eszű 1 k-s ZX81-en dolgozni. Valamilyen virtust vált ki: a „csak azért is megoldom" makacsságát.

Az érzelmekről a programozásra térve elmondom, hogy elsősorban a sallangmentességet és az információközlés átgondolt szükszavúságát tartom fontosnak.

A példa, amelyen állításaimat illusztrálom, egy szópóker játék. A gép gondol egy 4 vagy 5 betűs szót, aszerint, hogy a játékos milyen hosszú szó kitalálására vállalkozik, majd betűnként lehet találgatni. Ha a betű szerepel a szóban, de nem ott van, ahová a játékos kérdezte, akkor kijelzi, ha pedig minden stimmel, kiírja a betűt, majd végül az egész szót. A szót véletlenszerűen választja ki összesen 20 szavas (10 darab 4 betűs, 10 darab 5 betűs) szótárából.

(Aki ismeri a HT gép gyári kazettáját, tudja, hogy ott 16 k-ban elfelejtették megoldani a véletlenszerű választást. Így ha valaki másodszor ül le a HT géppel játszani, előre tudhatja, hogy a gép a „MALAC" szóval kezd.)

Végül hadd adjak néhány gyakorlati tanácsot arra, hogyan lehet egy-egy bájtot felszabadítani.

A gép a számokat valós számként kezeli, ezért 8 jegy pontossággal figyeli az egészeket is. A program írásakor a következő fogásokat alkalmazhatjuk.

5 bájtot lehet spórolni, ha
0 helyett NOT PI-t,
1 helyett SGN PI-t,
3 helyett INT PI-t írunk.

4 bájtot takaritunk meg, ha
A≧2 A<66, illetve A>128 esetén a számot kódként kezeljük, kikeresve a megfelelőt a táblázatból. Azoknak a számoknak, amelyek nem használatosak a táblázatban, még mindig nem a szokásos „A" alakját, hanem a VAL „A" alakját érdemes írni (3 bájt megtakarítás).

Így a program nehezebben követhető, de nem áll meg a 4-es, „nincs több hely a memóriámban" hibajelzésen.

VOTISKY ZSUZSA

```
 2 RAND
 4 INPUT N
 5 LET S=N*INT (RND*10)
 15 GOSUB (N=5)*70+300
 30 LET B$=A$(S+1 TO S+N)
 40 FOR C=SGN PI TO CODE "="
 42 PRINT AT C,NOT PI;C
 45 LET T=NOT PI
 50 FOR K=SGN PI TO N
 60 INPUT C$
 70 FOR L=SGN PI TO N
 80 IF C$=B$(L TO L) THEN GOTO
    (K=L)*200+(K<>L)*150
 90 NEXT L
 100 NEXT K
 110 NEXT C
 120 PRINT "KEZDJE UJRA"
 130 PAUSE 4E4
 135 CLS
 140 GOTO 4
 150 PRINT AT C,CODE "="+K;K
 160 GOTO VAL "90"
 200 PRINT AT C,K+SGN PI;C$
 250 PRINT C;" KERDESBOL KITALALTA"
 260 GOTO 120
 300 LET A$="PARTCICAHAMUMOZIUGAT
    SIMAREGENYILHULLFAGY"
 310 RETURN
 370 LET A$="KAROMFOGASPUSZIMAROK
    BORULDIVATCSIGABEREKAGYARTEREM"
 380 RETURN
```

16 k-s ZX81 gépekhez

CSILLAG

A játék célja a kép alsó részén megjelenő űrbázisok védelme úgy, hogy közben a védő üzemanyagkészlete és lövedékei ne fogyjanak el. A játék során ezek állásáról folyamatos viszszajelzést kapunk. Sikeres találat esetén jutalomban részesülhetünk belőlük.

A változók

Célkereszt: 16,D. Csillag: X,E. Üzemanyag: N. Lövedék: S. Bázis: 20,V.

Programtervezés

A program legfontosabb rutinjait a listában elhelyezett REM sorok jelzik. Az értékadó utasítások a 30–70 sorokban találhatók. A főprogram a 2–6. rutinokig tart és tartalmazza az RND generálást, az irányítást, a feltételvizsgálatokat (üzemanyag-regisztrálás, robbanás) és a törlést.

A program kimentése

Egy direkt GOTO 4000 utasítással menthetjük ki a programot úgy, hogy betöltés után RUN-nal indul.

A programot Radácsy Tivadar 8. osztályos tanuló készítette.

```
  1 REM * * * * * *
  2 REM *         *
  3 REM * CSILLAG *
  4 REM *         *
  5 REM * * * * * *
  6 REM
  7 PRINT AT 4,5;"5=BALRA,";AT
6,5;"8=JOBBRA,";AT 12,5;"0=TUZ."
  8 PAUSE 4E4
 10 CLS
 30 REM *ERTEKADAS*
 40 LET K=0
 45 LET N=15
 50 LET S=5
 55 LET I=0
 60 LET U=3
 65 LET D=14
 70 LET X=1
 75 LET A$=" *"
100 REM *GRAFIKA*
105 PRINT AT 0,7;"▀▀  ▀▀  ▀▀
 ▀▀"
110 PRINT AT 20,5;"▐▌      ▐▌
  ▐▌"
120 PRINT AT 21,3;"▟▙▟▙  ▟▙▟▙
 ▟▙▟▙"
130 PRINT AT 19,15;5
140 REM *RND*
150 LET R=INT (RND*4)
180 LET E=(R*5)+8
209 IF N<0 THEN LET N=0
210 LET E=E+INT (RND*3)-1
211 IF E>=31 THEN LET E=31
212 IF E<=0 THEN LET E=0
215 LET I=0
216 PRINT AT X,E;" ";AT 16,D;"
 "
220 LET X=X+INT (RND*2)
222 LET N=N-.1
225 IF N<=0 THEN GOTO 3000
230 REM *IRANYITAS*
235 IF INKEY$="5" THEN LET D=D-
1
240 IF INKEY$="8" THEN LET D=D+
1
250 IF INKEY$="0" THEN LET I=1
255 REM *FELTETELVIZSGALAT*
257 IF X>=20 THEN GOTO 2000
258 PRINT AT X,E;A$;AT 16,D;"-+
-"
260 IF I=0 THEN GOTO 400
265 LET S=S-1
270 PRINT AT 19,15;S;" "
285 IF (E=D+2 OR E=D OR E=D+1)
AND (X=16 OR X=17) THEN GOTO 700
287 IF S<=0 THEN GOTO 3000
290 LET I=0
310 LET N=N-1
400 REM *TORLES*
410 PRINT AT X,E;" "
420 GOTO 210
700 REM *UZEMANYAG*
702 LET S=S+2
705 PRINT AT X,E;" "
710 PRINT AT X+2,E-1;"▀  ▀";AT X
-2,E-1;"▀ ▄ ▀"
715 LET N=N+5
825 IF N>31 THEN LET N=31
830 FOR G=0 TO INT N
840 PRINT AT 12,INT G;"▀"
850 NEXT G
860 PRINT AT 10,13;"▀▀▀▀"
870 FOR I=0 TO 50
875 NEXT I
880 PRINT AT 10,13;"          "
883 PRINT AT 12,0;"
 "
885 PRINT AT X+2,E-1;"   ";AT X
-2,E-1;"   "
886 LET X=1
890 GOTO 125
2000 REM *ROBBANAS*
2050 PRINT AT 20,V;"▀  ▀";AT 21
,V;"▀ ▄ ▀"
2100 LET V=V+7
2110 IF V>24 THEN GOTO 3000
2115 LET X=1
2120 GOTO 125
3000 REM *BEFEJEZES*
3010 PRINT AT 2,9;"*JATEK VEGE*"
3012 IF N<=0 THEN PRINT AT 5,7;"
SAJNOS,NAFTA=0."
3015 IF S<=0 THEN PRINT AT 5,7;"
SAJNOS,LOVEDEK=0."
3020 FOR I=0 TO 75
3030 NEXT I
3050 PRINT AT 10,4;"ISMET ? (BAR
MI=IGEN/N=NEM)"
3060 LET O$=INKEY$
3065 IF O$="N" THEN STOP
3070 IF O$="" THEN GOTO 3060
3080 RUN 10
3999 REM *KIMENTES*
4000 SAVE "CSILLAG"
4010 RUN
```

HT–1080Z iskola-számítógéphez

KÍGYÓS JÁTÉK

```
10 REM      **********************
20 REM      **                    **
30 REM      **  KIGYOS JATEK      **
40 REM      **                    **
50 REM      ** Csirmaz Laszlo     **
60 REM      **                    **
70 REM      **********************
80 REM
90 CLS
100 PRINT@262,"KIGYOS JATEK":PRINT:PRINT
110 FORI=1TO500:NEXTI
120 REM **  Hanggeneralas  **
130 DIMHH(3):FORI=0TO13:READHH(I):NEXT
140 DATA8,8,8,8,8,8,15,8,16,16,16,8,50,8
150 REM **  A jatekszabalyok  **
160 CLS
170 PRINT"A bal oldalon ulo jatekos neve:"
180 INPUTA$
190 PRINT"Es a jobb oldalon ulo neve:"
200 INPUTB$
210 CLS
220 PRINT"     Kedves "A$"!"
230 PRINT"Te a 'W','A','S','Z' gombokkal"
240 PRINT"iranyitod a Kigyot fel, balra,"
250 PRINT"jobbra (ll. le.)":PRINT
260 PRINT"     Melyen tisztelt "B$"!"
270 PRINT"Te a 'P','L','+' valamint a '-'"
280 PRINT"gombokat hasznalhatod.":PRINT
290 PRINT"     A Kigyo nem keresztezheti"
300 PRINT"sem sajat magat, sem az ellen-"
310 PRINT"fel kigyojat, es nem utkozhet a"
320 PRINT"falba sem.Az veszit,aki ezekat"
330 PRINT"szabalyokat nem tartaja be."
340 PRINT:PRINT"Ha mehet, nyomj le egy gombot!";
350 X$=INKEY$
360 IFINKEY$=""THEN360
370 REM **  A palya szele  **
380 CLS:PRINTCHR$(151)STRING$(38,131)CHR$(171);
390 FORI=64TO896STEP64
400 PRINT@I,CHR$(149);:PRINT@I+31,CHR$(178)
410 NEXT
420 PRINT@960,CHR$(181)STRING$(38,176)CHR$(186);
430 X1=15:Y1=27:X2=48:Y2=27:U1=2:U2=2
440 L=8
450 SET(X1,Y1):SET(X2,Y2)
460 L=L+1
470 REM **  Indul a jatek...  **
480 U=PEEK(14349)AND142
490 IFUTHENU=4+(U)127)+(U)7)+(U)3)ELSEU=U1
500 ONU60TO510,530,550,570
510 IFU1=3THENU1=1:Y1=Y1-1ELSEY1=Y1+1
520 GOTO580
530 IFU1=4THENU1=2:X1=X1+1ELSEX1=X1-1
540 GOTO580
550 IFU1=1THENU1=3:Y1=Y1+1ELSEY1=Y1-1
560 GOTO580
570 IFU1=2THENU1=4:X1=X1-1ELSEX1=X1+1
580 IFPOINT(X1,Y1)THEN1=1:GOTO740
590 SET(X1,Y1)
600 U=(PEEK(14370)ANDB8)+(PEEK(14348)AND1)
610 IFUTHENU=4+(U)63)+(U)15)+(U)7)ELSEU=U2
620 ONU60TO670,690,650,630
630 IFU2=1THENU2=4:Y2=Y2-1ELSEY2=Y2+1
640 GOTO780
650 IFU2=2THENU2=3:X2=X2+1ELSEX2=X2-1
660 GOTO780
670 IFU2=4THENU2=1:Y2=Y2+1ELSEY2=Y2-1
680 GOTO780
690 IFU2=3THENU2=2:X2=X2-1ELSEX2=X2+1
700 IFPOINT(X2,Y2)THEN1=2:GOTO740
710 SET(X2,Y2)
720 GOTO460
730 REM ** Utkozes tortent! **
740 FORJ=0TO13+OUT31,J:OUT30,HH(J):NEXT
750 FORJ=1TO250:NEXT:FORJ=1TO10
760 FORH=1TO80:NEXT
770 IFI=1THENRESET(X1,Y1)ELSERESET(X2,Y2)
780 FORH=1TO80:NEXT
790 IFI=1THENSET(X1,Y1)ELSESET(X2,Y2)
800 NEXT
810 FORJ=8TO1000:NEXT
820 REM ** Ki a nyertes? **
830 CLS:IFI=2THENX$=A$ELSEX$=B$
840 PRINT@74,L"pontta!"
850 X$=X$+" nyert":L=LEN(X$):M=269-L/2
860 PRINT@M,STRING$(L+8,"*");
870 PRINT@M+64,STRING$(L+8,"*");
880 PRINT@M+128,"**"STRING$(L+4,32)"**";
890 PRINT@M+192,"**  "X$"  **";
900 PRINT@M+256,"**"STRING$(L+4,32)"**";
910 PRINT@M+320,STRING$(L+8,"*");
920 PRINT@M+384,STRING$(L+8,"*");
930 FORI=1TO300:NEXT
940 PRINT@832,"Megegyszer Jatszotok (I-N)?";
950 X$=INKEY$
960 X$=INKEY$
970 IFX$="I"THENPRINT" I";:GOTO880
980 IFX$()"N"THEN960
990 PRINT" N";
1000 END
```

A játékot két személy játszhatja. Mindketten egy-egy pontot irányítanak négy-négy gomb segítségével (W, A, S, Z, P, L, +, .). A vonalak nyoma megmarad, beléjük ütközni nem szabad.

A programutasítások eléggé specifikusak, hogy a játék gyorsabb legyen. Például a 49Ø-es utasításban az IF U THEN az IF U = Ø THEN végrehajtását gyorsítja. A 48Ø és 6ØØ-as sorokban közvetlenül a klaviatúra-mátrixot olvashatjuk le; így a játékosok egyszerre nyomkodhatják a gombokat.

CSIRMAZ LÁSZLÓ

A Novotrade Rt. videojáték-pályázatának eddigi eredményei

A Novotrade Rt. 1983 januárjában videojáték-ötleteket keresve, pályázatot hirdetett. A pályázatra mintegy 1300 pályamű érkezett. Ezek közül a Novotrade szakemberei a videojáték-piacot jól ismerő külföldi szakértők bevonásával kiválasztották a 300 legjobbnak ígérkező ötletet, és felajánlották azokat külföldi partnereiknek.

Piackutató tevékenységüknek köszönhető, hogy a mai napig mintegy 20 játékötlet programozása kezdődött meg. Hét játék teljesen elkészült; ezeket külföldi forgalmazók már meg is vásárolták. Némelyik közülük már kapható a boltokban Angliában és az Egyesült Államokban. A többi megjelenése néhány héten belül várható. Nagy örömünkre szolgál, hogy egyik játékunk, a Caesar the Cat, januárban az angol játékszoftver-slágerlista 4. helyére került.

A játékpályázatnak helyezettjei, díjazottjai nem voltak. A piac fogja megítélni, hogy kié volt a legjobb játék – ez az eladott mennyiségekből látszik. A szerzők „díja" egyenesen arányos az elért sikerrel: ők a Novotrade Rt.-vel kötött szerződés alapján, az eladott mennyiségek után részesednek szerzői jogdíjban.

Az elmúlt évben a videojátékok piaca és kereslete rendkívül nagy változásokon ment keresztül. A legnagyobb, meghatározó piac az Egyesült Államok: az itt észlelhető tendenciák általában hosszabb-rövidebb időn belül az európai és a távol-keleti piacokon is érvényesülnek.

Az egyik legnagyobb változás, hogy csökkent a kereslet a kizárólag videojátékok lejátszására alkalmas rendszerek iránt, viszont annál nagyobb érdeklődéssel fordult a közönség az otthoni számítógépek felé. Az előbbi legnagyobb kárvallottja az Atari cég, amely több, mint 600 millió dolláros veszteséggel zárta az évet, az utóbbi győztese pedig a Commodore, amely mikroprocesszor-eladásait tekintve a világ első helyére küzdötte fel magát.

A másik nagy változás, hogy az Egyesült Államokban az eddigi kazetta, ill. cartridge helyett leggyakrabban lemezen forgalmazzák a videojátékokat (mivel az otthoni számítógép tulajdonosoknak több, mint 90 százaléka rendelkezik lemezmeghajtóval). Ennek az az előnye, hogy a kazettán (cartridge-on) történő forgalmazáshoz szükséges, nagyon szigorú memóriakorlátok bizonyos mértékig feloldódtak, bár a lemezen forgalmazott programok is rendkívül tömörek.

Nagy változás állt be a játékok minőségében is. Egy évvel ezelőtt a tipikus játék kétdimenziós volt, rendkívül leegyszerűsített grafikával, és történetei általában változatok voltak arra a témára, hogy a valamely oldalról egyenes vonalban érkező ellenséget gyorsan le kell lőni. Azóta a közönség sokkal többet vár el a videojátékoktól, és kihasználva az otthoni számítógépek adta lehetőségeket, a játékok szerzői ki is elégítik a megnövekedett igényeket.

Az idén Las Vegasban rendezett Consumer Electronics Show-n bemutatott játékokra talán a hatalmas mértékben fejlődött grafikai megjelenítés volt a legjellemzőbb; szinte dot-szintű tervezést megkövetelő minőségű grafikai ábrázolást alkalmaznak. Majdnem minden játék háromdimenziós: hatalmas, reklámgrafikára emlékeztető perspektívákat használnak. A grafikai hatást fokozza, hogy kevert színekkel dolgoznak, multiplied sprite-okat mozgatnak igen gyorsan, amivel szinte rajzfilmszerű hatást érnek el. Jellemző a kétirányú scroll (egy-egy játék vagy több, egymás után következő jelenetet tartalmaz, vagy a scroll segítségével juthat a játékos a játék történetének egyik színhelyéről a másikra).

Igen nagy a jelentősége annak, hogy az idén bemutatott Commodore programok egy része *szoftver útján beszél*.

A játékok tartalmilag is fejlődtek: történelmi eseményeket, kalandregényeket dolgoznak fel. Az olimpiára készülve gyakorlatilag minden sportágra készült igen magas színvonalú program. S egyre nagyobb az érdeklődés az oktatásban használható, a tanítást érdekesebbé, szemléletesebbé tevő, a tanulást megkönnyítő programok iránt.

A Novotrade Rt. játékpályázata nem zárult le!

Továbbra is keresünk érdekes, izgalmas, eredeti, nem kultúrához kötött játékötleteket, de keresünk oktatásban használható programötleteket is.

Az ötleteket 1 db A4-es lapon kérjük. A lap egyik oldalán a játékötlet rövid, angol nyelvű leírása szerepeljen, másik oldalán pedig a képernyő tervezett jellegzetes állapotait bemutató, 6–8 jó minőségű, színes ábrát kérünk.

A beküldési határidő folyamatos. Előzetes értékelést minden hónap 30-ig készítünk.

A pályaműveket kérjük az alábbi címre *postán* eljuttatni:

Novotrade Rt. 1136 Bp. Fürst Sándor u. 24–26.

Bővebb információval munkatársaink: Boldizsár Ágnes és Kovács Éva (tel.: 530-022) szívesen állnak rendelkezésükre.

PÁSZTOR ESZTER

Robotron A–6401/02 típusú kisszámítógép-rendszerek

Vállalatunk közel két éve forgalmazza a ROBOTRON A–6400-as kisszámítógép család rendszereit.
A családnak két típusa ismert:
a K1620-as és a
K1630-as jelzésű mikroszámítógépekre épülő A–6401-es és az A–6402-es rendszerek.
Mindkét rendszerhez nagy választékban kapcsolhatók a különböző típusú és rendeltetésű perifériális készülékek.
A K1600-as mikroszámítógépek és az ezekre épülő A–6400-as kisszámítógépek alkalmazási lehetőségei és területei igen széles körűek.
A legfontosabbak az alábbiak:

Automatizált termelésprogramozás és irányítás
- egyedi és részfolyamatok vezérlése, gyártásfelügyelet
- gép- és ipari vezérlések
- folyamatirányítási adatok gyűjtése, mérési adatkoncentráció
  és digitális szabályozási feladatok
- hierarchikus vezérlőrendszerek irányítása

Kutatás-fejlesztés, ipar, egészségügy és oktatás terén a laboratóriumi és mérésautomatizálás feladatainak megoldásához
- labortechnikai rendszerek és egyedi készülékek vezérlése
- központosított és osztott mérésautomatizálás

Univerzális információfeldolgozó rendszerek területén
- vállalati és intézeti számítóközpontokban elvégzendő műszaki-tudományos és gazdasági számítások
- nagy adatfeldolgozó rendszerek részfeladatainak átvétele mint pl. terminálok, vonalkoncentrátorok, adatgyűjtő rendszerek
- önálló ügyviteli adatfeldolgozó rendszernek
- számítógépes hálózatokban
- kötegelt feldolgozás és
- a párbeszédes üzem valamennyi problémájának
  megoldására

Az A–6401/02 kisgépek párbeszédes használata lehetséges:
- a bankszakmában (számlaforgalom)
- a kereskedelemben
- az egészségügyben
- a rendőrségnél
- a közlekedésben
- a szállodaiparban
- különféle termelő üzemekben

A ROBOTRON A–6401/02-es kisszámítógépek rendszertechnikai felépítése
A központi feldolgozó egységet, a tármodulokat, a csatlakoztató vezérléseket és csatoló egységeket egy elektromosan és működésben szabványos kapcsolat ún. buszrendszer köti össze. A buszrendszeren történik a címek, adatok és vezérlő jelek továbbításán kívül a modulok energiaellátása is.

A buszrendszer három típusú buszból áll, amelyek rugalmas lehetőséget biztosítanak a különböző konfigurációk kialakításához:
- K–1630 tip. processzor-orientált busz
- K–1600 rendszerbusz
- MSZR (CM) busz

A K–1630 processzor-orientált busz segítségével aritmetikai műveletvégző (K–2062) csatlakoztatható a központi feldolgozó egységhez (K–2663), amely meggyorsítja a lebegőpontos műveletek végrehajtási idejét.
A K–1600 rendszerbuszon az adatok továbbítása aszinkron, a címek továbbítása szinkron. A rendszerbuszon keresztül történő kommunikáció formája valamennyi rákapcsolt készülék számára azonos, a „master-slave" elv alapján történik.
Az *egységes MSZR-busz* buszátalakítóval csatlakoztatható a rendszerbuszhoz.
A ROBOTRON K–1620 mikrogép a közepes teljesítményű modellek közé tartozik. A K–1620 az alapja a konfigurált A–6401-es adatfeldolgozó számítógéprendszernek, és a megfelelő tármodulokkal 32 k szóig bővíthető (1 szó = 16 bit)

A ROBOTRON K–1630 mikrogép
A mikrogép központi feldolgozó egysége lehetővé teszi real-time rendszerek, gazdasági és műszaki-tudományos adatfeldolgozó rendszerek, valamint többfelhasználós és multiprogramozott rendszerek kialakítását.
A K–1630 mikrogép az alapja a konfigurált ROBOTRON A–6402 adatfeldolgozó rendszernek. A megfelelő tármodulokkal 128 k szóig bővíthető.

INFORMÁCIÓTECHNIKAI VÁLLALAT

A ROBOTRON A–6401/02 kisszámítógép-rendszerek az MSZR rendszerrel szoftver- és hardver-kompatibilisek (a hardver-kompatibilitás ún. plugkompatibilitás, vagyis dugaszolható csatlakozó kompatibilis), így az MSZR rendszer átfogó rendszerszoftver, alkalmazási szoftver és perifériaválasztéka egyaránt felhasználható. A rendszer felépítése lehetővé teszi a csatlakoztatható perifériák és részegységek típusválasztékának bővítését.

A ROBOTRON A–6401/02 kisszámítógépekhez kapcsolható perifériális készülékek

K 5665	• Floppy diszk
K 6200	• Lyukszalagegység
CM 5400	• Mágneslemez
CM 5300	• Mágnesszalagegység
K 8912	• Display terminál
K 8931	• Univerzális terminál
VT 27060	• Sornyomtató
DARO 1157/264	• Mátrixnyomtató
	• Kazettás mágnesszalagegység

A ROBOTRON A–6401/02 kisszámítógépek szoftverkészlete teljes mértékben kielégíti a felhasználói igényeket. Fejlett operációs rendszerrel (MOOS 1600) rendelkezik, amelynek vezérlőprogramja moduláris felépítésénél fogva lehetővé teszi a különböző felhasználói követelmények legteljesebb kielégítését.

A vezérlőprogram legfontosabb szolgáltatásai:

- több felhasználó kiszolgálása
- multiprogramozás
- prioritásvezérlés
- időosztás (time-sharing)
- a futó felhasználói programokkal párhuzamos programfejlesztés a háttérben
- kötegelt feldolgozás (a job-vezérlés közvetett kommandókkal is lehetséges)
- felhasználói programok logikai összekapcsolása
- automatikus újraindítás hálózatkimaradás esetén, a hardverlehetőségeknek megfelelően
- időkezelés (dátum, óra, készülékfelügyelet)
- parancs módú feldolgozás
- készülékek és állományok lefoglalása bizonyos felhasználók számára
- fizikai készülékcímek hozzárendelése logikai készüléknevekhez
- a csatolt készülékek kezelésének vezérlése
- dinamikus tárkezelés és tárvédelem a ROBOTRON A 6402 kisgép K 2663 feldolgozó egységének használata esetén
- közvetlen és soros hozzáférésű állományok szervezése.

A felhasználók a programjaikat assembler nyelven, COBOL illetve FORTRAN IV. magas szintű programnyelvek egyikén írhatják. Megfelelő rendszerprogramok lehetővé teszik a CDL (Compiler Description Language) használatát.

Az elmúlt két év alatt vállalatunk mintegy húsz A–6401/02-es típusú kisszámítógép-rendszert értékesített.

Az értékesített rendszerek a népgazdaság különböző területeihez tartozó vállalatoknál, illetve intézményeknél működnek (pl. építőipar, járműgyártás, könnyűipar stb.). Vállalatunk arra törekszik, hogy a felhasználók részére komplex szolgáltatást biztosítson:

- garanciaidő alatti és utáni javítás, karbantartás
- alkalmazástechnikai tanácsadás
- felhasználói oktatás
- szoftvertámogatás
- rendszerkonfigurálási és telepítési tanácsadás
- a vállalat saját rendszerét háttérgépként szükség esetén a felhasználók rendelkezésére bocsátja.

Rövid és ravasz programok

Új kaleidoszkóp HT–1080Z iskolaszámítógépre

Az 1983. évi számban megjelent „Kaleidoszkóp" módosított változata, amely nemcsak rajzol, de zenél is. A programot készítette: Pócz Zoltán, Jászberény.

```
10 DATA 0,0,0,0,0,0,0,248,15,15,15,0,0,0,0,0
20 FOR A=0 TO 15:READ B:OUT31,A:OUT30,B:NEXT A
30 CLS:PRINT@84,"KEPERNYO GRAFIKA"
40 PRINT@192,"A RAJZOLAST EGY BILLENTYU LENYOMASAVAL LEHET LEALLITANI."
50 PRINT@320,"NYOMD MEG AZ 1,2,3 VAGY 4 BILLENTYUT!"
60 IF INKEY$="" THEN 60
70 IF INKEY$="1" THEN 240
80 IF INKEY$="2" THEN 280
90 IF INKEY$="3" THEN 320
100 IF INKEY$="4" THEN 110 ELSE 60
110 CLS
120 I=RND(127): J=RND(127):K=RND(47):L=RND(47)
130 FOR X=I TO J:OUT 31,0:OUT 30,2*X+1
140 FOR Y=K TO L:OUT 31,2:OUT 30,5.3125*(Y+1)
150 IF POINT(X,Y)=-1 THEN RESET(X,Y) ELSE SET(X,Y)
160 IF INKEY$<>"" THEN 190
170 NEXT Y: NEXT X
180 GOTO 120
190 CLS:OUT 31,0:OUT 30,0: OUT31,2: OUT 30,0
200 PRINT@70,"KOSZONOM A RAJZOLAST."
205 PRINT@860,"AKARSZ MEG CSENDBEN FIGYELNI?(I/N)"
210 IF INKEY$="I" OR INKEY$="N" THEN 220 ELSE 210
220 IF INKEY$="I" THEN 30 ELSE 230
230 END
240 CLS:FOR X=0 TO 127
250 FOR Y=0 TO 47
260 SET(X,Y)
270 NEXT Y: NEXT X: GOTO 120
280 CLS:FOR X=0 TO 126 STEP 3
290 FOR Y=0 TO 46 STEP 2
300 SET(X,Y)
310 NEXT Y: NEXT X: GOTO 120
320 CLS:FOR X=0 TO 126
330 FOR Y=0 TO 46
340 IF(X+1)/2-INT((X+1)/2)=0AND(Y+1)/2-INT((Y+1)/2)=0THEN350ELSESET(X,Y)
350 NEXT Y: NEXT X: GOTO 120
```

Milyen napon születtem?

Az alábbi program – Erdélyi Tibor küldte be – egy öröknaptár szerepét veszi át. A Commodore 64 személyi számítógépre készült program minden BASIC nyelvet ismerő számítógépen üzemel, ha a 2Ø-as és a 3Ø-as sorokat az adott géptípusnak megfelelően megváltoztatjuk.

```
10 REM ** NAPTAR [ERDELYI TIBOR] **
20 PRINT"□ EZ A PROGRAM MEGMONDJA, HOGY A BEUTOTT"
30 PRINT"DATUM MILYEN NAPRA ESETT!"
40 PRINT"A MEGADOTT DATUM 1582. UTANI LEHET CSAK!"
50 INPUT"KEREM A DATUMOT EEEE,HH,NN FORMABAN!";EV,HO,NAP
60 PRINT"□";EV;"■.";HO;"■.";NAP"■. ";
70 D=0:Z=1
80 IF HO>2 THEN D=INT(HO*.4+2.3):Z=0
90 X=365*EV+31*(HO-1)+NAP
100 X=X+INT((EV-Z)/4)-INT(3*(INT((EV-Z)/100)+1)/4)
110 X=X-D
120 Y=X-(INT(X/7)*7)
130 FOR I=0 TO 6
140 READ NN$(I)
150 NEXT I
160 PRINT NN$(Y)
170 END
180 DATA SZOMBAT,VASARNAP,HETFO,KEDD,SZERDA,CSUTORTOK,PENTEK
```


ADOK–VESZEK–CSERÉLEK

• **DIALOG–PÁRBESZÉDES TERMINÁL-KEZELŐ ELJÁRÁS.** Az eljárás (helyesebben eljáráscsomag) tetszőleges interaktív ügyviteli feladat összes terminálkezelési funkciójának megvalósítására lehetőséget ad a ma már szinte kizárólag alkalmazott képernyős terminálok lehetőségeinek kihasználásával. Elsődleges funkciója az adatbevitel megkönnyítése: ez „formázott" képernyőn, billentyűzés közbeni ellenőrzéssel és sokoldalú szerkesztési lehetőségek kihasználásával történhet. A szerkesztési lehetőségek kihasználásával előzőleg bevitt adatok helyben-javítására is mód nyílik. További szolgáltatásai: az input, illetve adatmódosítási funkcióhoz hasonló formázott output, valamint különböző üzenetek megjelenítése.

Az eljárás TAP–34 és VT20 mikroszámítógéphez készült változatai Assembler, az A6401 miniszámítógéphez készülő változatai COBOL nyelven írt programokból hívhatók. Utóbbiak az SZM–4, TPA 1140–1148, A6402 miniszámítógépeken is felhasználhatók.

A program ára, illetve bérleti díja egyedi megállapodás tárgya.

ÉPÍTÉSGAZDASÁGI ÉS SZERVEZÉSI INTÉZET
Budapest II., Csalogány u. 9.
Telefon: 152-390, Telex: 225138
Bővebb felvilágosítás:
ÉGSZI 27. Főosztály, Piackutatási csoport

• **SOROS ÉS INDEXES FILE-KEZELŐ PROGRAM.** Rendeltetése: Egy vagy több hajlékony mágneslemezen lévő fix – generáláshoz megadott – rekordméretű adatállomány (file) soros vagy indexelt, rekordszintű feldolgozásának biztosítása TAP–34 mikroszámítógépen.

A kezelt állomány maximum 19 hajlékony mágneslemezre terjedhet ki. A file-kezelő által generált állományok megfelelnek az IBM szabványoknak.

A rendszer szolgáltatásai:
– Állományok generálása
– Rekordszintű műveletek: soros olvasás, soros írás, kulcsos olvasás, kulcsos írás, kulcsos pozicionálás, kulcsos törlés
– Állományok vizsgálata, tesztelése
– Egyidőben 4 állomány feldolgozásának biztosítása.

Alkalmazási terület: Minden olyan TAP–34-re tervezett rendszerben, amelynek igényeit a géppel szállított eredeti file-kezelő nem elégíti ki.

A program ára, illetve bérleti díja egyedi megállapodás tárgya.

ÉPÍTÉSGAZDASÁGI ÉS SZERVEZÉSI INTÉZET
Budapest II., Csalogány u. 9.
Telefon: 152-390, Telex: 225138
Bővebb felvilágosítás:
ÉGSZI 27. Főosztály Piackutatási csoport.

• **6809 ALAPÚ GÉPEK** (EUROCOM, EXORCISER, EXORSET stb.) és a FLEX, valamint OS–9 rendszerek használóival felvennénk a kapcsolatot program- és tapasztalatcsere céljából. Érdeklődés a 364-853 számú telefonon.

• **ZX–81 ÉS ZX SPECTRUM** gépekre játék- és egyéb hobbiprogramok:
INTERBIT GMK
1631 Budapest Pf. 6.
Tel.: 856-028, 149-692
Kívánságra katalógust küldünk. Egyéni feladatra, ötletre programot írunk.

Kedves Olvasó!

Reméljük, hogy a rovatnak nemcsak olvasói és a feladatoknak szorgalmas megoldói, hanem írói is lesznek.

A feladatokat – melyek megoldását mindig a következő számban ismertetjük –, különböző korosztályok számára írjuk ki. A csoportok megkülönböztetésére római számokat használunk, a következőképpen:

I. 8–14 évesek
II. Középiskolások és szakmunkástanulók
III. Felsőfokú oktatási intézmények hallgatói
IV. Dolgozók

A megfejtéseket a rovat szerkesztőjének nevére küldjék: Neumann János Számítógéptudományi Társaság
Garádi János
Budapest V., Báthori u. 16.

Minden megfejtést jól olvashatóan, külön lapon kérünk. Minden lapon szerepeljen a feladat száma és a megoldó neve. A borítékban a neve és címe mellett mindenki tüntesse fel korosztályát és munkahelyét (iskoláját).

A legjobb, legötletesebb megoldásokat beküldőjük nevével együtt közöljük.

A kitűzésre szánt feladatokat és megoldásukat szintén a fenti címre várjuk.

A feladatok beküldési határideje: 1984. július 30.

Feladatok

1.

Számítsuk ki program segítségével, hogy hány darab természetes számnak (és melyek azok) van olyan tulajdonsága, hogy a szám egyenlő számjegyei faktoriálisainak összegével. Például $145 = 1! + 4! + 5!$ =
$= 1 + 1 \cdot 2 \cdot 3 \cdot 4 + 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 =$
$= 1 + 24 + 120$

Beküldendő a BASIC, FORTRAN, PASCAL program és a számok felsorolása.

(Ajánlott a II–IV. korosztály részére.)

2.

Készítsünk olyan programot, amely a következő feladat megoldására alkalmas:

Budapestről szeretne valaki külföldre telefonálni, és csak azt az információt közli, hogy melyik időzónába. Ha a számítógépben lévő programnak az ügyfél megmondja az időzónát és a magyar vagy a hívott időzóna dátumát, akkor a gép közli a hívott időzóna vagy a magyar dátumot. A dátum formája: év, hó, nap, óra, perc.

(1984 2 29 13 45)

Kérdés: A fenti feladat egyértelműen van megfogalmazva? Ha nem, akkor egészítsük ki, hogy a programot (BASIC, FORTRAN, PASCAL, PL/1) vagy az algoritmust el tudjuk készíteni!

Az időzónákat a közületi telefonkönyvben találhatjuk meg.

(Ajánlott az I–IV. korosztály részére.)

3.

Az alábbi programmal kapcsolatban a következő kérdésekre kell beküldeni a választ:

a) Milyen feladat(ok) megoldásánál tudjuk felhasználni a programot?

b) Hogyan kell kezelni? (Mit kell a gépnek megadni és hogyan?)

c) Lehet-e a programot úgy módosítani, hogy a futási idő csökkenjen? (Mit kell módosítani, ha lehet?)

(Ajánlott a II–IV. korosztály részére.)

```
10 DEFDBL A,B,C
20 INPUT A:CLS
30 B=INT(SQR(A)): A=A+2
40 IF(A/3)-INT(A/3)=0 THEN 30
50 FOR I=3 TO B STEP 2
60 C=A/I
70 IF C-INT(C)=0 THEN 30
80 C=A/(I+2)
90 IF C-INT(C)=0 THEN 30
100 NEXT I:PRINT A,: GOTO 30
```

Megoldások az előző számban közölt feladatokhoz

1. Péter és barátja az alábbi karaktereket gépelték be a személyi számítógépükbe:
Ø szám, . , + , – , jeleket, mert a zérust az előbbi formában is meg lehet adni a HT–1080Z, Commodore 64 stb. gépeken.

E fenti jeleken kívül az alábbi jelek begépelése után is Ø számot ír ki a különböző típusú számítógép:

Commodore 64 E→Ø
HT–1080Z !→Ø
} szám

2. A Hungarocamion csomagolói részlegében az alábbi algoritmus szerint érdemes dolgozni:

- Először súlyuk szerint sorba kell rendezni a csomagokat.
- A legnagyobb súlyúval kezdve, rendre az egység-konténerekbe kell rakni azokat.
- A ládákat (konténereket) sorszámmal látjuk el, és ezek a számok sorrendjében (1, 2, 3, ...) helyezkednek el.
- Minden csomag a lehető legkisebb sorszámú ládába kerül, amelybe még a 100 kg-os összsúly túllépése nélkül kerülhet.

A fenti algoritmus az esetek 9/11-ed részében a legjobb!

3. Berkó László játékprogramja az alábbi módon működik a HT–1080Z iskolaszámítógépen:

- A gépen a program indítása után megjelenő kérdőjelre adott számérték 1, 2, 3, lehet, mert különben ismételten megkérdezi a gép, és nem megy tovább.
- A program ezután egy ejtőernyős alakzatot jelenít meg a képernyőn, és ezt tudjuk a balra és a jobbra nyíllal vagy az 5-ös és 6-os számokkal irányítani.
- Ha a csoport egy véletlenszerűen készített felhőben megakad, akkor a képernyőn megjelenik a „BUMM" felirat.

Nyertesek

Az 1983. évi számban megjelent feladatokra határidőre beérkezett megfejtések közül hibátlan volt:

1. feladat
(I. és II. korosztály részére) 23 megfejtés
2. feladat
(III. és IV. korosztály részére) 15 megfejtés
3. feladat
(I. és II. korosztály részére) 29 megfejtés

Könyvutalványt nyertek:

I. korosztály: Fekete András, Komló, Gagarin úti általános iskola.

II. korosztály: Király László, Jászberény, Erősáramú Szakközépiskola; Jónás Róbert, Putnok, Gimnázium és Mezőgazdasági Szakközépiskola; Győri István, Szeged, JATE Ságvári Endre Gyakorló Gimnázium; Lieb Zsolt, Nyíregyháza, Bánki Donát Ipari Szakközépiskola; Polgár Sándor, Bp., Hámán Kató Közgazdasági Szakközépiskola; Krasznói Attila, Kecskemét, Katona József Gimnázium; Nagy Róbert, Kecskemét, Katona József Gimnázium; Görgényi Gábor, Miskolc.

III. korosztály: Faragó Csaba, Salgótarján, Pénzügyi és Számviteli Főiskola.

IV. korosztály: Nádasy Gábor, Bp.; Szabó László, Bp.; Szalontai István, Tatabánya.

A nyertesek részére a 150 forintos könyvutalványt a szerkesztőség postán küldi el.

Kabala kerestetik!

A μ Magazin szerkesztősége felhívással fordul a

Kedves Olvasókhoz,

hogy segítsenek egy, a lapot népszerűsítő

„totem figura"

megalkotásában, majd kiválasztásában.

A beküldött rajzokat, bábukat, fényképeket és más alkotásokat Magazinunkban folyamatosan bemutatjuk. 1984 szeptemberében (az iskolakezdést is ünnepelve) az olvasók szavazatai alapján közösen választjuk ki azt a figurát, amely találóan kabala bábukként és még sok más formában „képviseli" majd Magazinunkat.

A legtöbb szavazatot kapott alkotást díjazzuk.

A pályaművek beküldése folyamatos, a végső beküldési határidő 1984. augusztus 15.

Várjuk a pályázatokat!

Az olvasó írja

Nem panaszkodom, dicsekszem: körülbelül 200 levelet kaptunk az 1983. évi szám megjelenése után (beleértve a kitöltött kérdőíveket is). Engedjék meg, hogy néhány közös kérdésre, problémára egyszerre válaszoljak, és csak néhány levélre adjak név szerinti választ.

• Kevés helyen lehetett a lapot megvásárolni. Sajnos. A Posta Központi Hírlap Irodával tárgyalásokat folytatunk, hogy a lapot szélesebb körben terjesszék. Néhány ezer példányt mindig tartunk a Társaság központjában (Bp. V., Báthori u. 16.), ott folyamatosan kapható. Ezenkívül a Műszaki Könyvesboltban (Bp. VI., Liszt F. tér 9.), a Statisztikai Kiadó Vállalat Könyvesboltjában (Bp. II., Keleti K. u. 10–12.) és a SZÁMALK Könyvesboltban (Bp. XI., Szakasits Á. út 68.) lehet hozzájutni.

• Várjuk és kérjük a programokat. Ha valaki a forrásprogram-listát ki tudja nyomtatni, kérem, úgy küldje; sokat segít a szerkesztőségnek.

• Legszívesebben a rengeteg dicséretet közölnénk egymás után, de nem szeretnénk, ha a dicsőség a fejünkbe szállna. Azért írják, – bevalljuk, jólesik. A szokásos két oldalon inkább a kritikákat hozzuk.

• Olvasóink általában keveslik a hardver-cikkeket. Megpróbálunk segíteni...

• Sokan kérik, hogy ismertessük a gépi kódú programozást. Megtesszük.

Jámbor Lajos, Debrecen,

Lehel út 4.

Szeretnék néhány észrevételt tenni. Kicsit kevés a hardver. Remélem, ez a jövőben bővülni fog.

Még valami a ZX-ről: van egy bosszantó hibája, nagyon sokszor ledermed, befagy, elszáll, vagy nem is tudom, mit mondjak.

Van valakinek ötlete, hogy mit kell a befagyott ZX81-gyel csinálni, illetve mit kell tenni, hogy ne fagyjon be? Írja meg!

Krizsák László, Székesfehérvár,

Vörös Hadsereg út 94.

Egy kérésem is lenne. A ZX81-es gépi kódjának táblázata biztos sokakat érdekelne, tehát ha lehet, a közlését szeretném kérni.

Emlékül két programot kaptunk, amelyet próbálunk. A gépi kódú programírásról szeretnénk cikket közölni. Szerző kerestetik!

Tomka Miklós, Budapest,

Várvíz u. 4.

Keresek olyan valakit, aki egyrészt a TI 99/4A gépi szintű programozásában tudna segíteni (esetleg német vagy angol szakirodalom kölcsönzésével), másrészt keresek programcsere-partnert, aki esetleg olyan „fortélyokat" is el tudna mondani, amelyek nem szerepelnek a géphez adott kézikönyvben. (Természetesen ha többen tudnának segíteni, vagy többen akarnak programot cserélni, annál jobb.) Ezekhez kérném az Önök segítségét. Esetleg Önök nem tudnak Magyarországon hozzáférhető TI 99/4A könyvről?

Megpróbálunk mi is segíteni, de lehet, hogy az olvasók között akad a megfelelő társra.

Gál Péter, Budapest,

Hermann Ottó u. 8.

Hosszú leveléből csak részleteket tudunk közölni:

Én alig több, mint egy éve foglalkozom számítástechnikával. Azt hiszem, matematikai problémák egyszerű megoldhatósága vezetett el a gép szeretetéhez. Ezért is hiányolom néhány egyszerűbb, de látványos matematikai program leírását, esetleg rövidebb cikkekből álló sorozatot is el tudnék képzelni. Mikre gondolok? Például: konvergencia-vizsgálat, függvényábrázolás, prímszámok vizsgálata stb.

Mindenképpen hasznosnak tartanám egy feladatgyűjtemény elindítását – versenyen kívül. Rengeteg példával, mindenféle erősségűvel.

Nagy szükség lenne egy alap hardver-tanfolyamra is. Innen megérthető lenne az egész gép működése, esetleg a gépi kód is.

Javaslataival egyetértünk. Segítene? A küldött feladatokat átadtam az Agyafúrmánynak.

Szabó László, Aszód,

Szabadság tér 7.

1. Hogyan lehet inverzben írni a HT–1080Z-vel?

2. Milyen rövidített utasításai vannak a gépnek?

Kérdéseit átadtam az Agyafúrmánynak és az illetékeseknek. A programokhoz küldjön rövid leírást is; kezdők is vannak a világon.

Vődrös László, Komárom–Szőny,

Széchenyi u. 6.

Olvasónk azt írja: 66 éves, nyugdíjas paraszt. Én azt hiszem, hogy levelezőink doyenje. Köszöntöm és köszönöm az elektronikus szebszótárt leíró levelét, nagyon élveztem. Sajnos, megvalósították már, de mesterséges beszéddel nyelveket tanító mikroszámítógépet is árulnak. Az eszperantónak, ahogyan nevezi „hidnyelvként" való alkalmazásával Társaságunk Szövegfeldolgozási és humán alkalmazási szakosztálya foglalkozik. Szívesen látjuk tagként, keresse meg Titkárságunkat. A „Mi és a computer" című tv-adásnál együttműködünk. Remélem, mire kiadványunk az utcán lesz, az adás címét is sikerül magyarítanunk.

Kristófy Gyula,

főiskolai hallgató

A FORTH programozási rendszerről megjelenő cikkel kapcsolatban kérdezem, hogy hol lehet hozzájutni a témával foglalkozó magyar nyelvű szakirodalomhoz (szerző, cím, kiadó).

Azért akad problémám is. Mivel ZX81-tulajdonos vagyok, ki szerettem volna próbálni a közölt programokat. De csak szerettem volna.

A BEVETÉS nevet viselő program 210. sora, illetve a LÓVERSENY nevű program 60. sora hibás. Próbáltam megfejteni, hogy mi kellene a hibás sorok helyébe, de rövid idő után feladtam. Bosszant a dolog, hogy miért nem lehet egy programot hibátlanul leközölni. Sajnos ez más lapoknál is többször előfordul (például Rádiótechnika, illetve az Ötlet BIT–LÉT melléklete).

FORTH könyv rövidesen megjelenik a Műszaki Könyvkiadó gondozásában, beszámolunk róla. A hibákat sajnáljuk, az 1984. 1. számban a 210-es sor javítását közöltük. A 60. sor jó, nézze meg még egyszer a programlista feletti magyarázatot. Kérem, írja meg címét, mert a boríték elkallódott. Elnézését kérem.

Hegedüs Zoltán, Hódmezővásárhely,

Károlyi M. u. 70.

Jó lenne, ha indulna egy sorozat, amely egy kisgép építését tenné lehetővé. A legnagyobb gondunk, hogy nincsenek kapcsolási rajzaink. Természetesen bő magyarázattal.

Indul. Küldje el a programokat.

Barna Zsolt, Ózd,

Münnich Ferenc u. 3/b.

A Computer Szaküzlet aktuális választékából rendszeresen (árakkal) ismertetniük kellene egy-egy géprendszert, például alapgép, kazettás periféria stb. (Nem közületi pénztárcához igazítva az ismertetést.) Mindezek mellett meg kellene adniuk, mit tud a gép feladatméretre vonatkozóan. Vagy hogy milyen adatterület marad például egy kis számítást végző program mellett.

A nem közületi pénztárcához igazítás és a szaküzletek árai között egyelőre antagonisztikus az ellentmondás. Sajnos.

Bálint Balázs János, Baja,

Duna u. 10.

Nagy tett, hogy megindították ezt a magazint! Szívesen olvasnék minél többet a mesterséges intelligencia kutatásokról, a heurisztikáról. Nagyon örülnék, ha a humán dolgok is kellő súlyt kapnának: számítógépes grafika, számítógépes verselőrendszerek, számítógépes költészet (például betű, illetve képversek, parodizálás), számítógépes zeneszerzés, társalgó programok stb.

R. D. Laing, Gubancok és „Tényleg szeretsz...?" köteteinek anyagát akár számítógép is írhatta volna, akárcsak a 10 évvel ezelőtti Tandori-verseket, vagyis elképzelhető egy filozofikus, nonszensz, abszurd számítógépes költészet is, mivel az „ilyesmi" bizonyos mértékben algoritmizálható is.

Mi is nagyon szeretnénk nem numerikus témájú, „humán" tartalmú számítástechnikai cikkeket; szívesen teret adunk ezeknek az írásoknak. Levele nyomán remélem kedvet kap valaki „irodalmi" témájú írás beküldéséhez.

Szinyei Béla,

tervező

Településtervező lévén, rengeteg lehetőségünk volna mikroszámítógép használatára. Számítunk arra, hogy Önöktől ebben a tevékenységben is támogatást fogunk kapni.

Kérem írjanak – népszerűen – munkájukról, hátha a leírás valakit számítógép alkalmazására ösztönöz, esetleg támogatást ad a feladat megoldásához.

Nádasy Gábor, Budapest,

Páva u. 32/b.

Javaslataiból néhányat felsorolunk, foglalkozni fogunk velük.

– Sorozat az operációs rendszerek ismertetésére CP/M, DOS stb.

– Egy rövid ismeretterjesztő rovatot lehetne szentelni egyéb µP-os alkalmazásoknak (például video, gépjármű, fotó, zene, űrkutatás stb.)

– Belföldi, külföldi árak rendszeres ismertetése. (Támadás az erkölcstelen hazai árképzés – számítógép árképzés – ellen.)

Küldjön cikkeket, programokat, ötleteket.

Dévényi Zsolt, Miskolc,

Egyetemváros

Én is szívesen részt vennék a szoftverpiacon egy komplett operációkutatási programcsomaggal, amely BASIC programnyelven íródik, és VC–20 gépre készül. A programok átírását más típusú gépekre is vállalom.

Szívesen felvesszük a programot az Adok–Veszek–Cserélek rovatba. Küldje be hirdetés formájában.

Stomp László, Rakamaz,

Ady E. u. 27.

Jómagam a katonaságnál találkoztam először személyi számítógéppel, amelynek SHARP PC–1500 volt a márkája. Ez négyszínű golyóstollas (!) nyomtatóval volt ellátva és folyadékkristályos kijelzővel. A közel egyévi ismerkedés és barátkozás eltolta az érdeklődési körömet a számítástechnika felé.

Kellemes hallgatni, hogy a sorkatonai szolgálat ilyen szerencsés pályamódosítást is eredményez. Jó parancsnokai lehettek.

Horváth Katalin, Eger,

Gerl Mátyás u. 4.

Programjaimat ZX–Spectrum gépre szoktam készíteni, a személyi számítógépek közül ez érdekel a legjobban. Legutóbb erre a gépre készítettem bioritmus-programot.

Rövid programjaim is vannak a ZX–Spectrum képernyő-grafikájának bemutatására.

Egyetértek Sajni József (Székesfehérvár) véleményével: igenis érdekelné az olvasókat a számítógép történetével foglalkozó cikksorozat, rovat.

A programot várjuk, hátha jelentkezik valaki, aki a rövidítésére vállalkozik. Az 1984. 1. számunkban már közöltünk technikatörténeti cikket, ezután is fogunk.

Zsák Tibor, Budapest,

Galgóczy u. 27.

Arra gondolok, hogy mi sem egyszerűbb (és szükségszerűbb), mint a most várhatóan beinduló „szoftver-nagyüzem" során a nagyvállalatok tapasztalataira támaszkodva kialakítani egy országosan egységes programozói stílust, szabványajánlást és dokumentálási módszert.

Például: az egyes klubok, iskolák és amatőrök között megindul a program csere-bere. Ha géptípus-eltérés vagy más egyéni igény miatt a program új tulajdonosa javítani szeretne a programon, nem mindegy, hogy egy kommentek nélküli, GOTO-kkal feleslegesen felaggatott és utólagos átgondolatlan „toldozgatással" teletűzdelt programot kell-e veritékes munkával megérteni, vagy egy világos, egyszerű (és dokumentációval rendelkező) programot kell javítani.

Egyre több lesz a különböző okokból létrejött pályázat, és nem irigylem azt a zsürit, amely vagy ötven különböző pályaművet bírál el programozástechnikailag, ha a programozók különböző koncepciók szerint, enyhén szólva „művészi ihletésből" dolgoztak, és persze megfelelő dokumentáció nélkül. (A programforráslista önmagában nem elegendő dokumentáció!)

Leveléből hosszabban idéztünk, mert igen fontos témával foglalkozik. A továbbiakban országos kódrendszert és nyilvántartást ajánl és azt, hogy egy programozó új program írása előtt győződjék meg arról, hogy nem csinált-e már valaki hasonlót.

Én egy kicsit szkeptikus vagyok, az a tapasztalatom, hogy egy intézeten belül is nehéz megvalósítani egységes technológiát és dokumentációs rendszert – sok-sok előnye ellenére, de kell!

Várjuk olvasóink véleményét.

Bankó János, Békéscsaba,

Majakovszkij u. 31.

Szeretném javasolni, hogy a HCC keretében indítsanak számítógép-építő tanfolyamot, amelyben részenként bemutatnák egy számítógép megépítési részleteit, kapcsolási és szerkezeti rajzokkal támogatva. Javaslom, hogy közöljék gyári számítógépek (ZX81, HT–1080Z) szerkezeti, kapcsolási rajzát, leírását.

A HCC, mint az 1984. 1. számban meghirdette, számítógép-építést indít; megpróbálunk központilag gondoskodni alkatrészről. Másik ötlete a gyártóktól függ. Elképzelhető, hogy ha elolvassák javaslatát, vállalkoznak egy ilyen cikk írására is.

Szedmer József, Tatabánya I.

Nedermann F. u. 15.

Szükségesnek tartanám olyan sorozat megjelentetését, amelyben a gépépítés komplett folyamatát ismerhetnénk meg. Hogy konkrétan mire gondolok? A szükséges kapcsolási rajz leírása alkatrész-szükséglettel, a klaviatúra, az adatrögzítő (például magnetofon) kialakítása, illetve kapcsolata a központi egységgel, és jó lenne a beszerzési lehetőség feltüntetése is.

Lásd: Bankó Jánosnak adott válaszomat.

Cserép Zoltán, Hajdúnánás,

Kossuth u. 22.

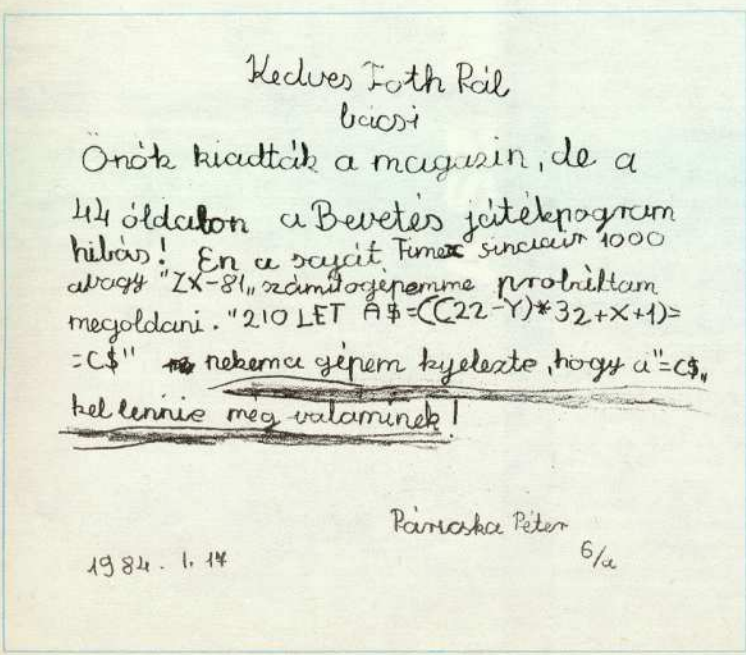
Elemző levelének néhány részlete:

Tetszett a „Mi a személyi számítógép?" című cikk. Kimondottan laikusokhoz szólt, az ún. tudományos ismeretterjesztést jól szolgálta. Az Iskola – számítógép rovatban a pályázat keltette fel érdeklődésemet. A tanfolyamot hasznosnak tartom, bár a stílusa – főleg a bevezető részé – nem nagyon tetszett. Programozástechnika rovatuk telitalálat. Nekem ez tetszett a legjobban. Az Ember–gép-kapcsolat már csak témája miatt sem érdekel túlságosan. Az Agyafúrmány és Játékprogramok rovatok – legalábbis szerintem – sok olvasójuknak, köztük nekem is, szereztek örömteli perceket!

Foglalkozott még az iskola-számítógéppel, az AIRCOMP–16-tal, a magas piaci árakkal. Levélben válaszoltam.

Köszönöm levelezőink bizalmát, az ötleteket, javaslatokat, programokat. Várom a cikkeket is. Szívélyes üdvözlettel:

KOVÁCS GYŐZŐ



HT–1080Z és HT–2080Z személyi számítógép tulajdonosok figyelem!

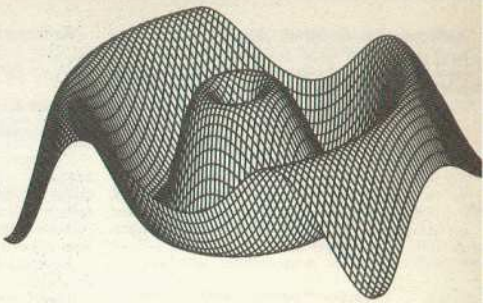
Kibővített konfiguráció = sokrétű alkalmazhatóság + nagyobb teljesítmény = professzionális felhasználás

Bővítési lehetőségek KONTASET dobozban ESZR kártyákon:

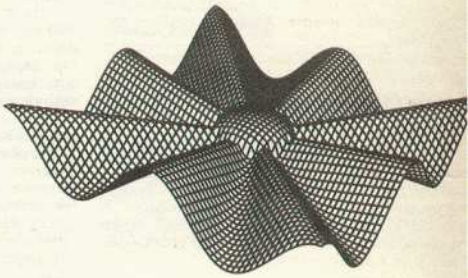
- RAM 8K–32K, lapozással 64K lépésekben
- PROM 8K–16K
- Grafika: 256 × 256 pontraszter megjelenítés
  512 × 256 hisztogram megjelenítés
- Nyomtató
- Floppy
- Analóg ki- és bemenetek programozott és/vagy DMA átvitellel
- Digitális párhuzamos ki- és bemenetek programozott és/vagy DMA átvitellel
- V 24, 20 mA, RS 422 csatlakozási felület
- Lokális hálózat kialakítás (56 Kbit/s vagy 2 Mbit/s átviteli sebességgel)
- Egyéni igényekhez kialakított hardver és szoftver illesztő felület

Részletes ismertető és érdeklődés:

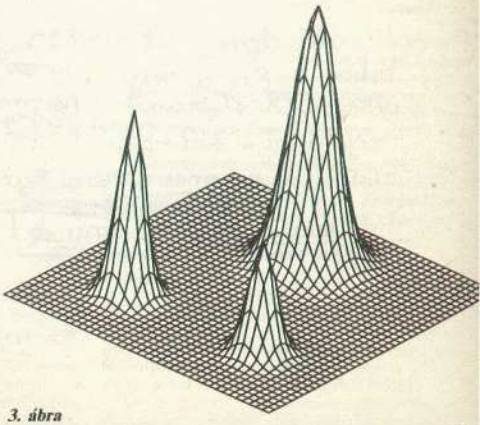
MTA Atommag Kutató Intézete
Debrecen,
Pf. 51.
4001



1. ábra



2. ábra



3. ábra

ISMERKEDJÜNK A SZÁMÍTÓGÉPES GRAFIKÁVAL!

Az ember természetes igénye gondolatainak rögzítése mások és saját maga részére is. Gondolatait szívesen továbbfejleszti, átalakítja, de ezt nem tudja a maga által kitűzött cél eléréséig szakadatlanul végezni; szüneteket tart, amelyekhez ismét az elért eredmények rögzítése szükséges. Ennek egyik legkedveltebb módja a vizuális rögzítés, amely szövegben, számokban, grafikában valósul meg.

A mondanivaló lerajzolása ősi közlési forma. Gondoljunk csak a barlangrajzokra! Nem véletlen. A rajz az információkat tömören tartalmazza, ugyanakkor általánosabban érthető, mint az írott szöveg.

Szellemi céljainkat megvalósító gondolataink továbbfejlesztésének fontos és lassan mindennapi eszköze a számítástechnika. Miért ne használnánk ezt az eszközt az információk tömör közlésére, a grafikai ábrázoláshoz?

A számítógépes grafika a nálunk számítástechnikai szempontból fejlettebb országokban is most kezd elterjedni. Ennek oka egész egyszerűen az, hogy a „tömör" képi információ igen sok részinformációra épül, amihez megfelelő méretű számítógép-kapacitás és nem is akármilyen feldolgozási sebesség szükséges. A mikroelektronika térhódításával azonban ezek az akadályok is elhárulnak. Érdemes tehát elkezdeni játszani, gyakorolni a számítógépes grafika lehetőségeit, hogy azokon okulva, a gyakorlatban is kamatoztatni tudjuk tudásunkat.

Ki ne ismerné a felismerés örömét, amikor négyzetháló papíron az $y = A \cdot x^2$ függvényt ábrázolva látja, hogy A értékétől függően miként változik a parabola alakja?

A következőkben néhány függvény számító- és rajzgéppel készült ábráját mutatjuk be. Olyan szépek, hogy akár díszítőelemként is használhatnánk őket.

Az *1. ábra* a

$$z = 7 \cdot \sin\left(\frac{\pi}{7} \cdot |\sqrt{x^2+y^2}\right) + 7$$

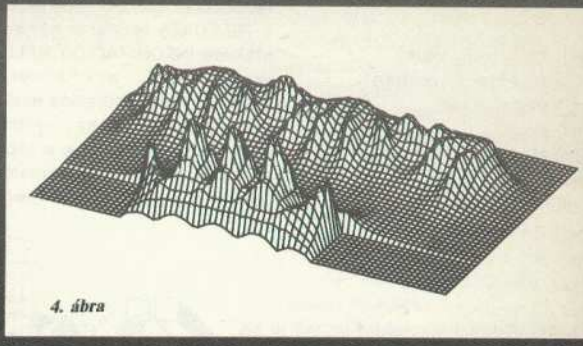
összefüggés alapján készült. A *2. ábra* már kissé bonyolultabb. A

$$z = \frac{1}{2} \cdot |\sqrt{12{,}25 - x^2 - y^2},$$

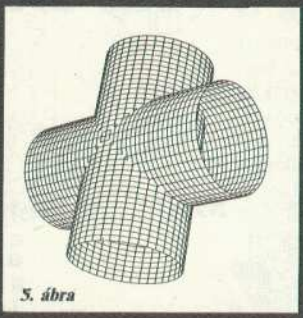
ha $x^2 + y^2 \leqq 12{,}25$

és a $z = \frac{x}{5} \cdot \cos\left(2 \cdot \pi \cdot \frac{y}{x}\right)$,

ha $x^2 + y^2 > 12{,}25$



4. ábra



5. ábra

6. ábra

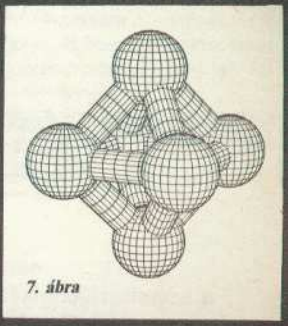
összefüggések alapján készült. A *3. ábra* egy lényegesen bonyolultabb függvényt ábrázol.

A *4. ábra* nagyon is gyakorlati alkalmazást mutat. Segítségével egy tervező vizsgálta meg, hogy az általa kiválasztott és megtervezett elrendezésű lámpatestekkel milyen megvilágítást kap egy adott területen. Vessük össze ezt az ábrát az *1.* és *3.* ábrával is. Ha valaki hozzájut egy kicsit jobb grafikus képernyőhöz, érdemes kísérleteznie.

A számítógépes grafika fő célja, hogy olyan gazdaságos ábrázolási lehetőséget nyújtson, amely az eddigiekben csak lassan és drágán volt elérhető. Ehhez a megfelelő eszközökön kívül olyan programok is kellenek, amelyek képesek ezt a célt megvalósítani. A számítástechnikai kutatási feladatok jelentős részét képezik a grafikával kapcsolatos kutatások.

A *5.*, *6.* és *7. ábra* a CAD (számítógéppel segített tervezés) folyóirat 1983. évi számából való. Nem nehéz belátni, hogy a láthatósági, az áthatási problémák megoldásának milyen gyakorlati jelentősége van. Ma már vannak olyan eszközök, amelyek ezeket a képeket néhány másodperc alatt jelenítik meg.

A számítógépes grafika szokásos módszere: tárolt elemekből néhány



7. ábra

grafikai művelet (például másolás, elmozgatás, méretarány-változtatás) alkalmazásával új ábra felépítése.

Ha sikerül hozzájutnunk színes grafikus képernyőhöz, kipróbálhatjuk a színek egymás melletti harmonizálását, különböző színes foltok egymás melletti hatását is.

Amint a felsorolt példák is mutatják, a számítógépes grafika hamarosan teljesíteni fogja azokat a követelményeket, amelyeket a rajzi információközléstől elvár az ember. És ezt olyan gyorsan és olyan változatos formában teszi, amilyet kézi technikával nem lehet elérni.

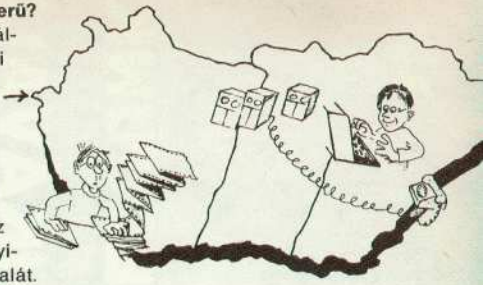
JANCSÓ FERENCNÉ

Ilyen még nem volt a számítástechnikában!

Nem kell beruházni, lehet bérelni! Bérelhet kész programcsomagokat!

Mi az, amiben a Scitel teljesen újszerű?

Magyarországon először alakult vállalat a komplett számítástechnikai rendszerek bérletezésére! Célunk a TELEDATA rendszer bázisán kialakított INFORMÁCIÓS HÁLÓZAT létrehozatala, az általánosan használt kommunikációs eszközök felhasználásával: az információáramlás meggyorsítása, a naprakész adatbank elősegíti a termelésirányítást, a helyes döntések meghozatalát.



bemutatkozik a Scitel?

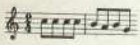
mi a Scitel?

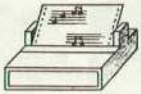
Számítástechnikai Fejlesztő Leasing Leányvállalat az SzKI 1984-ben megalakult új leányvállalata

Szki Scitel leányvállalat

c gységes
d atbankok,
s zámítógépes
i nformáció,
h agy rendszerek,
g azdaságos bérletezés

Professzionális személyi számítógép rendszereink:



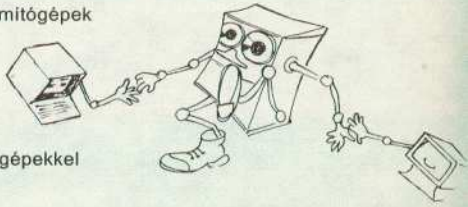


a kottaolvasástól
a takarmánykeverék-optimalizálásig
a vállalatirányítástól
a termelésirányításig
számtalan területen felhasználhatók!

nem ért a számítástechnikához? mi értünk hozzá, megtanítjuk!

Komplett alkalmazói rendszereinkről szívesen adunk részletes tájékoztatást az érdeklődőknek!

A professzionális személyi számítógépek használata
nem igényel szakembert
gyors és pontos információ
nagyobb termelékenység!



A professzionális személyi számítógépekkel lépésenként vagy egyszerre is megoldhatja számítástechnikai problémáit.

Széles körű alkalmazási lehetőség. – Modulrendszerű kialakítás.

Kapcsolódási lehetőség nagy számítógépekhez. – Bővíthetőség.

Mit kínál a Scitel **Scitel**?

MO8X professzionális személyi
PROPER 8 számítógépeket és az
PROPER 16 ezekre épülő
komplett alkalmazói rendszereket

Vállalati gazdálkodás
készletnyilvántartás és -gazdálkodás,
főkönyvelés és költséggazdálkodás,
munkaügy és bérelszámolás,
termeléselőkészítés
és műszaki nyilvántartás.

Mezőgazdaság
takarmánykeverék-optimalizálás
éves és középtávú tervezés

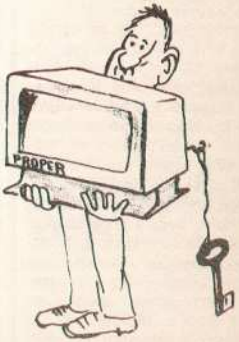
Szövegkezelés és feldolgozás
információ-visszakeresés
matematika, statisztika
elektronikai tervezés-automatizálás
kép- és hangfeldolgozás
lineáris programozás
grafikus programok
kémiai, labortechnikai programok
komplett kórházi rendszerek
raktári ügyvitel
postai adatgyűjtés
pénzügyi szervezés

TELEDATA rendszerek az információs hálózat kialakításához
MICROTEST – mikroprocesszoros fejlesztő rendszer a vezérlő, irányító és követő rendszerek tervezésének meggyorsításához

R16 számítógéprendszereken alapuló vállalati információs és termelésirányítási rendszerek
Teljesen kompatibilis az ESzR sorozat számítógépeivel, továbbá az ESzR első sorozatának ESz 1020-as és annál nagyobb modelljeivel, valamint az IBM 360-as és 370-es család tagjaival. Számítási kapacitása és bemeneti/kimeneti teljesítménye alkalmassá teszi hatékony felhasználását mind önálló számítógéprendszerben, mind számítógéphálózatok részeként.
Az R16 számítógép rendszerösszeállítása lehetőséget nyújt helyi és távoli, valamint kötegelt és párbeszédes üzemmódban történő működésre.
Az új típusú többprocesszoros, virtuális tárral megnövelt teljesítményű, gyors számítógépen egyidejűleg több program futtatható. Programozáskor a felhasználó igényétől függően hat programnyelv közül választhat.
Operációs rendszere a virtuális tárkezelést biztosító ESzR DOS–3, de képes a számítógép az IBM DOS/VS használatára is. Áramköri megoldásai korszerű hibajavítást és rendkívül egyszerű kezelhetőséget biztosítanak.

TANÁCSADÁS
A Scitel az SzKI 15 éves gyakorlatával a háta mögött az ÖNÖK rendelkezésére áll tanácsadással, igény szerinti rendszertervek kidolgozásával, üzembe helyezéssel, garanciális karbantartással.
Rendszereink használata különös szakértelmet nem igényel, kezelése könnyen, gyorsan elsajátítható.

Típuskonfigurációink
kulcsrakész kölcsönzését
rövid szállítási határidővel vállaljuk!



1015 Budapest
Donáti u. 35–45.
Tel.: 260-000
Telex: 22-5381
MNB: 207–11214
1251 Budapest
Pf.: 19

Minden érdeklődőnek
szívesen ad
részletes tájékoztatást:
Galambos Sándorné.

Észrevételek a PASCAL nyelvkönyvekről

A PASCAL nyelvet megtanulni szándékozók és a számítástechnikával foglalkozók örömére gyors egymásutánban két PASCAL programozási nyelvkönyv is megjelent magyar nyelven. Tekintettel a nyelv világviszonylatban megfigyelhető előretörésére, amely éppen a mikroszámítógépek és a személyi számítógépek kategóriájában a legszámottevőbb, igazán nem túlzás két magyar nyelvű PASCAL nyelvkönyv megjelenése.

Az egyik Gordon Erzsébet, Körtvélyesi Gézáné, Sós István és Székely Zoltán munkája. Címe: „PASCAL Programozási Nyelv" (a továbbiakban Gordon könyv). A másikat a PASCAL nyelv megalkotói, K. Jensen és N. Wirth írták, Mikulás Zoltán fordította. Címe: „A PASCAL programozási nyelv (Felhasználói kézikönyv és a nyelv formális leírása)" (a továbbiakban Jensen könyv).

A két könyv lényegileg különbözik egymástól. Jensen könnyed és elegáns, Gordon aprólékos. Jensen rendszeresen megemlíti a PASCAL nyelv eltéréseit más magas szintű programozási nyelvektől – Gordon feltételezi, hogy az olvasó más magas szintű programozási nyelvet nem ismer, így a hasonlóságokról és különbségekről egyáltalán nem szól.

Jensen példaprogramjai a programozásban jártas olvasó számára is tanulságosak. Legnagyobbrészt teljes programok, így az olvasó maga is lefuttathatja őket gyakorlásképpen. Ha ezt megteszi, több szintaktikai hibát talál. Kíváncsiságból megnézheti a Jensen könyv eredeti kiadását: ebben az összes program szintaktikailag és szemantikailag helyes. Sajnos a Műszaki Könyvkiadó nem követi azt a világon egyébként mindenütt bevett szokást, hogy a programokat a számítógép által kiadott, hibátlan lista közvetlen kinyomtatásával tegye közzé. Pedig a programok hagyományos módszerrel történő szedése mindig hibákat eredményez. A nyomtatási képben a kezdők számára előnyös lett volna az előre deklarált azonosítókat a programban definiáltaktól megkülönböztetni.

A Gordon könyv programozási példái mesterkéltek, és ami még ennél is nagyobb baj, mindig csak programtöredékek, így kipróbálni nem lehet őket, szintaktikai és szemantikai hibái nem deríthetők ki. Gordon könyvében még az alapszavak nyomtatási képe sem tér el az azonosítókétól, ami nyomdatechnikai szempontból az elképzelhető legigénytelenebb kivitel.

A PASCAL nyelvvel való megismerkedésem óta mindig lenyűgöz eleganciája és egyszerűsége. Talán az egyetlen dolog, ami megkeseríti a kapcsolatunkat, hogy a nyelv nincs megfelelően definiálva, legalábbis a két magyar nyelvű könyvben és az ezekben hivatkozott művekben nincs.

A Jensen könyv második része „Jelentés (A PASCAL nyelv formális leírása)", amely sajnos még annyira sem definiálja a PASCAL-t, mint az első rész, a „Felhasználói kézikönyv", de legalább az kiderül, hogy Jensen ezt tekinti minden hiányossága ellenére a PASCAL definíciójának.

A Gordon könyv sokszor leírja azt, hogy a szabványos PASCAL-ról beszél, de egyetlenegyszer sem árulja el, hogy melyik az a szabvány, amely a PASCAL nyelvet definiálja, és hogy ez hol található meg. A naiv olvasó azt hihetné, hogy talán a PASCAL megalkotóinak jelentését tekintik szabványnak, de ezt cáfolják azok a lényeges ellentmondások, amelyek a Jensen és a Gordon könyv között találhatók (lásd később). Azt is gondolhatnánk, hogy Gordon és szerzőtársai alkották meg a PASCAL szabványt, hiszen a II. rész címe „A szabványos PASCAL programozási nyelv leírása", de ezt cáfolja egyrészt az, hogy ezt sehol nem jelentik ki, másrészt az, hogy a könyv első része több helyen ellentmond a második résznek.

Ezek után megállapíthatjuk, hogy a szabvány kérdése egyáltalán nincs tisztázva. Ez pedig azért is szomorú, mert a Gordon könyv második részének három fejezetéből kettő a szabvánnyal foglalkozik – más szempontból is gyönge színvonalon. Itt meg kell említenem, hogy a PASCAL előtt FORTRAN-t tanítottam, ami ilyen tekintetben elkényeztetett, mivel az IBM kézikönyvek tökéletesen, egyértelműen és minden részletében definiálták a FORTRAN-t.

Felsorolok néhányat a Jensen és a Gordon könyv között fennálló, már említett ellentmondásokból.

– A Gordon könyv szerint az „=" operátor állhat tömbök, rekordok és halmazok között is (123. oldal, 139. oldal, 170. oldal). A Jensen könyv ezt tagadja a 112. és a 151. oldalon is.

– A Gordon könyv szerint kulcsszavakban és azonosítókban a kis- és nagybetűk felcserélhetők (203. oldal), Jensen szerint nem (141. oldal).

– A Jensen könyv a 88. oldalon felsorol három megkötést az eljárás- és függvényparaméterekkel kapcsolatban. A Gordon könyv a 220. oldalon ezeket a megkötéseket kihagyja, azaz betartásukat nem követeli meg.

A következőkben a Jensen könyvvel kapcsolatos észrevételeimet sorolom fel.

– Nagy kár, hogy a D függelékben található szintaxis csak a nyelv context free tulajdonságait tartalmazza, a context sensitive szintaktikai szabályokat nem. Ez lényegében használhatatlanná teszi ezt a függeléket. Még nagyobb baj, hogy ezt a hiányosságot nem árulja el, így a kezdő programozókat becsapja, azt a látszatot keltve, mintha az itt felsorolt szintaktikai szabályok betartása elegendő lenne szintaktikailag helyes programok készítéséhez (például a D függelék valós kezdőértéket is megengedne a *for* ciklusban).

– Nem derül ki, hogy az A...F függelékek a 15. fejezethez, vagy az első részhez tartoznak-e. Ez azért lényeges, mert így nem dönthető el, hogy a függelékekben lévő dolgok az első részben tárgyalt standard PASCAL-ra vonatkoznak, vagy csak a 14–15. fejezetben tárgyalt implementációra, a PASCAL 6000–3.4-re.

– A blokk szintaxisát bemutató ábra (14. oldal és 124. oldal) hibás, mert nem tükrözi, hogy egy blokkban egyszer és csak egyszer, és csak az adott sorrendben szerepelhetnek a „LABEL", „CONST", „TYPE", „VAR" alapszavak. A bal oldali függőleges vonalról hiányoznak a lefelé mutató nyilak. Ezt az apróságot azért említem meg, mert nagyon sok PASCAL leírás veszi át ezt az ábrát, például a Gordon könyv is.

– A Jensen könyv tárgymutatója nagyon gyenge.

– A be- és kivitel nincs minden részletében tisztázva.

– A magyar nyelvű kiadásban a fejezeteket átszámozták. Kár, hogy a fejezetekre való hivatkozásnál ez nem történt meg (például 22. oldal).

– Az itt felsoroltakon kívül még vagy 20 értelemzavaró sajtóhibát találtam.

Néhány észrevétel a Gordon könyvvel kapcsolatban:

– Sok, a PASCAL-hoz nem tartozó dologról ír fölöslegesen. Például szerintem teljesen fölösleges a 43. oldal, a 21–22. ábra, a 4.2.2 szakasz és a 3. rész 2.2.1 szakasz.

– A dinamikus változók és a mutatótípus nem érthető meg. Ilyen röviden teljesen fölösleges tárgyalni.

Az első rész első fejezete a számítógépekkel és a programozással foglalkozik, sok pongyolasággal tarkítva. Például a gépi kódú utasításokat igen primitív elemi műveletnek nevezi, holott már a 8 bites μP-nak is vannak nagyon is komplex gépi kódú utasításai, és akkor még nem is beszéltünk egy 32 bites μP-ról vagy egy mikroprogramozott nagygépről.

– Sok ellentmondás található a Gordon könyvben. Példaként csak egyet említek: a 38. oldalon azt állítja, hogy REAL típusú számban kötelező a tizedespont szerepeltetése, míg a 203. oldalon ezt tagadja.

– Sok a pongyola, zavaros megfogalmazás. Például a 212. oldalon azt állítja, hogy az állománytípust úgy lehet elképzelni, mint egy határozatlan hosszúságú tömböt. Mindenesetre ez rossz elképzelés.

A Gordon könyv nagyon hasznos része az implementációk leírása. Itt legfeljebb azt sajnálhatjuk, hogy több implementáció kimaradt. Hiányolom az ausztrál PASCAL-t. A legnagyobb csalódást viszont az okozta, hogy a Magyarországon leginkább elterjedt PASCAL implementációt, a Hisoft cég PASCAL 4T fordítóprogramjának ZX-Spectrumon futó változatát, a HP4S-t nem tárgyalják. Azt hiszem, Magyarországon többen használják a HP4S implementációt, mint az összes ismertetett többit. A HP4S nagyon korrekt, kevés megszorítást tartalmazó PASCAL implementáció, így igazán kár, hogy éppen ez maradt ki.

Ezek után melyik könyvet ajánljam a PASCAL nyelvvel megismerkedni szándékozóknak?

Azt hiszem, érdemes lenne mind a kettőt megvásárolniuk; kiegészítik egymást. Sajnos kapni nem lehet egyiket sem. Talán a Gordon könyvből nagy utánjárással még egy-egy elcsíphető. Nagyon remélem, hogy a kiadók megszánják szegény tanulni vágyó embereket, és mielőbb megjelentetik a második kiadást. Még nagyobb örömmel töltene el, ha az itt felsorolt hibák legalább egy része a második kiadásban már nem fordulna elő.

DR. DROZDY GYŐZŐ

Játékprogram pályázat

Nyertesek, helyezettek, díjazottak

1. díj:
Egy Junoszty televízió.
Nyerte: Tóth Péter IV. oszt. tan. (78 pont),
Kecskemét, Katona J. Gimn.
Gazdaasszony – Macska – Egér – Sajt

2. díj:
Egy MK kazettás magnetofon.
Nyerte: Bruckner Nándor IV. oszt. tan. (74 pont),
Sopron, Széchenyi I. Gimnázium.
Kígyó

3. díj:
10 db AGFA kazetta.
Nyerte: Rátkai István I. oszt. tan.
(71 pont),
Budapest, Fazekas M. Gimn.
Űrvadászat

4. helyezett:
Tőkey Gyula II. oszt. tan.
(64 pont),
Budapest, Fazekas M. Gimn.
Autóverseny, Űrjáték, Számkitalálás

5. helyezett:
Pereczes Zsolt II. oszt. tan.
(61 pont),
Várpalota, Lovászi L. Gimn.
Jégcsap

6. helyezett:
Nagy Ottó I. oszt. tan. (49 pont),
Nyíregyháza, 107. sz. Mező Imre Szakmunkásképző.
Derby

7. helyezett:
Szilágyi Éva III. oszt. tan.
(46 pont),
Budapest, Szerb Antal Gimn.
Barkochba

8. helyezett:
Phillipovics Ákos IV. oszt. tan.
(39 pont),
Budapest, Arany János Gimn.
Ágyú

9. helyezett:
Tóth Frigyes III. oszt. tan.
(35 pont),
Eger, Közgazdasági Szakközépiskola.
Várjáték
és Uzonyi Tamás IV. oszt. tan.
(35 pont),
Budapest, Eötvös J. Gimnázium.
Barkochba

Közönségdíjak:
Tóth Péter (61 szavazat),
CASIO zsebszámológép;
Pereczes Zsolt (40 szavazat),
könyvutalvány;
Bruckner Nándor (33 szavazat),
Rátkai István (27 szavazat);
Szilágyi Éva (16 szavazat).

Hírek, érdekességek

Mikroszámítógép-kölcsönzés Magyarországon

A Magyar Tudományos Akadémia Műszerügyi és Méréstechnikai Szolgálata több mint 25 éve kölcsönöz műszereket. 1983 végén új szolgáltatásként bevezette mikroszámítógépek kölcsönzését is. Az induló géppark 5 db ROSY 80 típusú (ROLITRON PJT) és 4 db MO8X (SZKI) hazai gyártású mikroszámítógépből áll. Mindkét típus nem csupán önálló mikrogépként, hanem a megfelelő interfész segítségével automatikus mérőrendszerek központi vezérlőegységeként is alkalmazható. A havi kölcsönzési díj a beszerzési árnak kb. 4 százaléka. A rendeltetésszerű használat során bekövetkezett esetleges hibákat a kölcsönző cég szervizszolgálata díjtalanul kijavítja.

FELIX CUB

Új mikroszámítógéppel jelentkezett a legnagyobb román számítógépgyár, a FELIX. Az Intel 8080 funkcionális megfelelőjén alapuló FELIX CUB mikrogép a professzionális kategóriába tartozik. Operatív tára 16–64 kbájt közötti, a csak olvasható tára pedig 2–16 kbájtos. A rendszer beépítve tartalmaz egy mini- vagy normál hajlékonylemezes tárolót is.

Operációs rendszere a már szinte világszabványnak tekinthető CP/M, annak számtalan előnyével. Például a programnyelvek széles választéka áll a felhasználó rendelkezésére (BASIC, COBOL, FORTRAN IV, Pascal stb.), és gazdag szerviz- és alkalmazási programkészlete van.

TPA-QUADRO

A Központi Fizikai Kutatóintézet új professzionális személyi számítógéppel jelentkezett. A szokatlan, de elegáns formatervezésű TPA-QUADRO a sikeres TPA–8 számítógépcsaládhoz tartozik. AM 2900 bitszelet mikroprocesszorra épül, operatív tára 128 k 12 bites szó. Perifériákkal jól kiépíthető, például hajlékonylemezes tárolókon kívül kazettás és Winchester lemezegységek is csatlakoztathatók hozzá.

Programkompatibilitása a PDP–8 gépcsaláddal eleve széles programkészletet biztosít számára. Ezen túlmenően már a megjelenéskor a felhasználók rendelkezésére áll például a COS/H ügyviteli, az RTS/H valósidejű operációs rendszer, továbbá az IL 128 párbeszédes és a WPS–8 szövegfeldolgozó rendszer.

Új lengyel mikroprocesszor-család

Lengyelországban is elkezdték 1983 második felében a 8 bites mikroprocesszorok gyártását: megjelent az MCY 7880 jelű, amely az Intel 8080 funkcionális megfelelője. Konfigurálását a gyakorlatban az alábbi táblázatnak megfelelően végzik.

Az európai szocialista országokban – Románia kivételével – már mindenütt gyártanak mikroprocesszorokat. Az Intel 8080 funkcionális megfelelőjét gyártják kronológiai sorrendben a Szovjetunióban KR580IK80A, Csehszlovákiában MHB 8080, hazánkban a Mikroelektronikai Vállalat 8080 és amint említettük, újabban Lengyelországban MCY 7880 néven. A Motorola 6800 megfelelőjét Bulgária SZM 601 néven és a Z80 funkcionális megfelelőjét az NDK gyártja U880 néven.

Típus	Megnevezés	Gyártó ország	A funkcionális analóg típusa (Intel)
K573PO1	8 kbájt EPROM	SU	2708
MCY 7114	4 kbájt RAM	PL	2114
MCY 7102	8 kbájt RAM	PL	2102
K580JW53	3 programozható számláló	SU	8253
MCY 745414	8 szintű vektorizált megszakító	PL	8214
MCY 7855	4 párhuzamos B/K csatorna	PL	8255
MCY 7851	1 soros (V.24) csatorna	PL	8251

MIKROFIN–KESSI szerződés

Az angol MIKROFIN cég szerződést írt alá a bolgár KESSI-vel, amely szerint valamennyi szocialista országban a KESSI forgalmazhatja az angol cég mikrogépeit. A MIKROFIN már leszállította Bulgáriának a P és F sorozatú mikrogépeinek négy-négy modelljét. Ezek közös jellemzője, hogy elemről üzemeltethetők, hordozhatók, mindössze tenyérnyi nagyságúak, és csupán tárméretben különböznek (a csak olvasható tár 16–32 k között, az operatív tár 8–32 k között változik). Áruk – teljesítményükhöz képest – a belföldi termékek árszínvonala alatt van.

A készülékek hordozhatósága lehetővé teszi, hogy az adatokat közvetlenül keletkezési helyükön gyűjtsék, és bizonyos időközönként (például este) közönséges telefonvonalon továbbítsák a számítógéphez.

A KESSI már tárgyalásokat folytat 600 mikrogép jugoszláviai exportjára, és megkereste a hazai illetékeseket is.

Csehszlovák mikroszámítógép

Elkészült az első csehszlovák gyártmányú, az MSZR 2. sorozatba tartozó SZM 50/40–1 kategóriájú gép. Mint ismeretes, még 1982 májusában bemutatták hazánkban az ebbe a teljesítménykategóriába tartozó szovjet SZM–1800 mikroszámítógépet. Nincs ugyan az MSZR sorozatba hivatalosan bevizsgálva, de teljesítményét és architektúráját tekintve kb. ezeknek felel meg a szintén 1982-ben bemutatott lengyel PSPD–90 és az 1983-ban megjelent román CUB mikroszámítógép is.

Ezek a gépek – mint az SZM kategóriajelzésben szereplő 1-es szám mutatja – az Intel 8080-nak szocialista országokban gyártott funkcionális megfelelőin alapulnak; a csehszlovák gép természetesen az ott gyártott MHB 8080-at tartalmazza.

Az SZM 50/40–1 operatív tára max. 64 kbájt, a csak olvasható tára 16 kbájt. Csatlakoztatható hozzá több hajlékonylemezes tároló és nyomtató. A felhasználó több operációs rendszer közül választhat. Rendelkezésére áll a valósidejű ER 4, a többfelhasználós MUOS és a multiprogramozható MIKROS.

Az SZM 50/40–1-ből az első szállítmány a hazai felhasználókhoz várhatóan ez év második felében kerülhet.

A 3. Mikroszámítógép Sakkvilágbajnokság

Lezajlott Budapesten, a SZÁMALK székházában, 1983. október 13–19. között, a Magyar Sakk Szövetség és a Híradástechnikai Tudományos Egyesület szervezésében, a Mikroprocesszor '83 nemzetközi konferencia keretében, több országos intézmény (köztük a Magyar Tudományos Akadémia, a Neumann Társaság és a Központi Statisztikai Hivatal) patronálásával. A szervező bizottság vezetője dr. Lindner László, a technikai bizottságé dr. Joó Gábor, az értékelő zsűrié Rajna György, a technikai zsűrié dr. Rétsági György, a legfelső, nemzetközi zsűrié dr. Benjamin Mittman professzor (USA) volt. Részt vett a szervező bizottságban többek között Frederic Friedel (NSZK) és Pierre Nolot (Franciaország). A hétfordulós, svájci rendszerű versenyen 8 ország 18 számítógépe és programja vett részt.

Az eredmény

A világbajnoki címet Elite A/S szerezte meg (Fidelity, USA), 6 ponttal. A további sorrend: Mephisto X (Hegener+Glaser, NSZK) 5 pont, 29 Buchholz-pont; Novag X (Hongkong) 5/27,5; Super Constellation (Novag) 5/25; Prestige (Fidelity) 4,5; Chess 2001 (Intelligent Software, Anglia) 4/26; Gedeon X (Magyarország) 4/12,5; Chess 2001 X (ISW) 3,5/27; Mephisto Y (H+G) 3,5/24; Mephisto Excalibur (H+G) 3/29,5; Constellation (Novag) 3/24; Sensory 9 (Fidelity) 3/23,5; Superstar X (Scisys, Hongkong) 3/22; Micromurks II. (Hamburgi Egyetem) 2,5; Logichess 2,2 (Koppenhágai Egyetem) 2/26; Chessmaster (Mikroelektronik Erfurt, NDK) 2/21; 65 Cyrus X (ISW) 2/15,5; Labirint 64 (Románia) O.

Egy kis magyarázat és értékelés

A svájci rendszerű verseny párosítása fordulónként úgy történik, hogy azonos vagy megközelítően azonos pontszámú versenyzők kerüljenek szembe egymással (a párok pontszámai közötti különbözetek összege minimális legyen), azzal, hogy kétszer ugyanazok nem mérkőzhetnek. Itt még az a megszorítás is szerepelt, hogy ugyanannak a cégnek a készülékei egymással nem játszhattak. A végső sorrend megállapításánál az azonos pontszámúak között a Buchholz-egyetem alapján döntöttek: összeadták az ellenfelek pontszámát, amelyekkel szemben mérkőztek. Ez az egyébként igazságos és bevált rendszer mindazonáltal nem ment a véletlenszerűségektől.

Különdíjat tűztek ki a legjobb, kereskedelmi forgalomban levő készülék, valamint a legjobb amatőr program részére. Az előbbit a Sensory 9, az utóbbit a Mikromurks nyerte. A 1–3. helyezett tiszteletdíjat kapott, amelyet a VIDEOTON Számítástechnikai Gyár, az Elektromodul Vállalat és a Mikroelektronikai Vállalat ajánlott fel. A különdíjakat a Számítástechnikai Koordinációs Intézet, illetve a Neumann Társaság adta. A felajánlott szép és értékes díjak is mutatják, hogy a számítógépes sakk iránt a témában érdekelt vállalatok, intézmények részéről komoly érdeklődés mutatkozik.

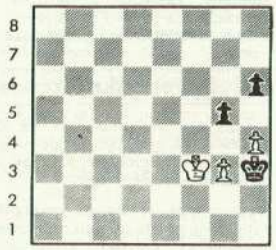
A Fidelity számítógépek ezúttal is, noha csak hajszállal, de a legjobbnak bizonyultak. Alighanem véletlen, hogy ezúttal az Elite ért el a Prestige-nél jobb eredményt, amelynek pedig – azonos program mellett – nagyobb sebességű a processzora (3 MHz-cel szemben 4 MHz). A Mephisto és a Novag teljesítménye ugyancsak kiváló. A Mephisto programok is megközelítően azonosak, ahogyan a cég közli, a Mephisto III program különböző versenyverzióiról van szó. Külön kiemelendő, hogy a Novag két különböző kísérleti programja azonos magas pontszámot ért el. Levy vállalatának, az Intelligent Software-nek teljesítménye ezúttal valamivel szerényebb volt. A számítógépek óriási fejlődésére jellemző, hogy a Scisysnek a legutóbbi világbajnok Mark V.-nél feltehetően erősebb kísérleti készüléke ugyancsak szerény helyezést ért el. Örvendetes volt a szocialista részvétel, noha nekünk ezen a területen természetesen sok a tanulnivalónk. Örültünk a hamburgi és koppenhágai egyetemen készült kitűnő amatőr programok szereplésének.

A következőkben a verseny néhány érdekes mozzanatának bemutatásával igyekszünk a sakkszámítógépek és -programok játékának jellegzetességeit, tudásuk jelenlegi szintjét érzékeltetni. Nemegyszer bámulattal adózunk nekik, olykor értetlenséggel nézzük hibáikat, de keressük és rendszerint meg is találjuk ezek magyarázatát. Legalábbis azt hisszük.

Végjátékelmélet

Dan Spracklen, hiába ő a férfi a családban, nem hitte volna, hogy számítógépe, a hírneves Prestige megnyeri a következő végjátékot. Valóban, döntetlent tarthatott volna világos, ha így játszik.

Constellation – Prestige



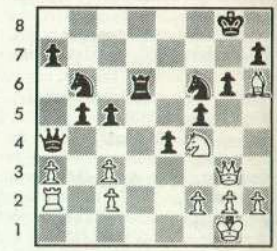
Világos lép

1. hxg5, hxg5 2. g4! (Elméleti tudás kell ennek felismeréséhez. 2. Kf2? veszít: 2. – g4! 3. Kf1, Kxg3 4. Kg1, Kf3 5. Kf1, g3. 6. Kg1, g2 és sötét nyer.) *2. – Kh4 3. Kf2!, Kxg4 4. Kg2, Kf4 5. Kf2, g4 6. Kg2, g3 7. Kg1!* és ez az állás döntetlen, a gyerekek is tudják. De a számítógép nem így számol. Ebben a változatban gyalogot veszt, s ezt a gyalogot meg is mentheti *1. h5?* lépéssel, ezért ezt húzza. (Ez egyébként valójában a 70. lépés.) Aligha lehet a szemére vetni, hogy nem volt képes kiszámítani: most már akkor is elveszti a játszmát, ha mindkét fél gyalogja egymás után alakul át vezérré. Ebben a pillanatban még csak néhányan láttuk azok közül, akik körülálltuk a táblát. Odahívtuk Sidney Samole-t, a Fidelity elnök-vezérigazgatóját, és megmondtuk neki: Prestige nyer. Nem akarta elhinni. Pedig ez történt: *1. – Kh2 2. Kg4, Kg2 3. Kf5, Kxg3 4. Kg6, Kf4 5. Kxh6, g4 6. Kg7, g3 7. h6, g2 8. h7 g1V+9. Kf7, Va7+10. Kg6, Vd4* és sötét nyert. Dan Spracklen utóbb bevallotta, nem értette az egészet. Miért nem lépett világos 10. Kg8-at? Akkor neki örökös sakkot kell tartania, vagy bemegy az ellenfél gyalogja is vezérnek. A számítógép azonban jobban tud sakkozni programjának alkotójánál. Hiába 10. Kg8, mert Kf5 11. h8V, Kg6! és a sarokban álló vezér nem tudja védeni a mattot. Ez az állás is ismert a végjátékelméletben, lehet, hogy Prestige előtt is?

Az elnézés

Az út egyáltalán nem volt rögöktől mentes Fidelity és Elite előtt. Az ötödik fordulóban világossal igen nehéz helyzetbe került.

Prestige – Mephisto X



Világos lép

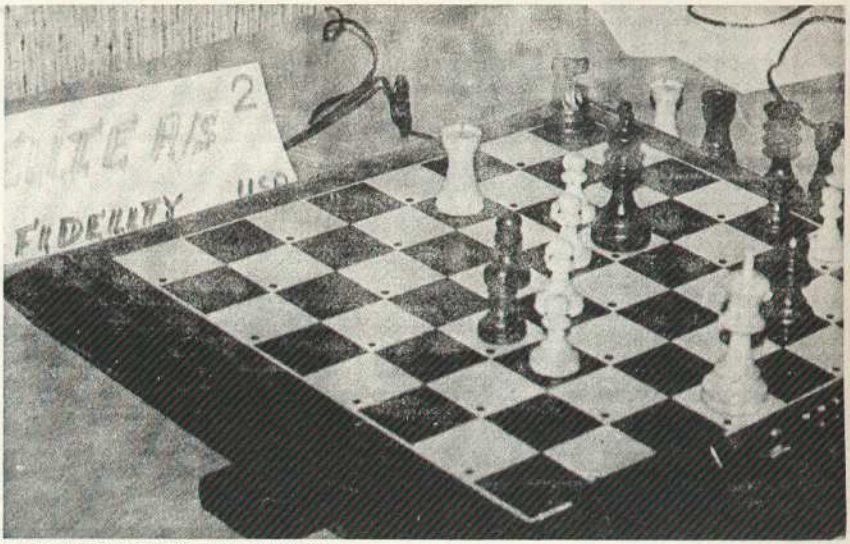
36. lépése ebben a hadállásban *Hd5??* volt, vagyis ütésbe állította huszárját. Kathe Spracklen kétségbeesett. „Ezer és ezer teszt során ilyesmi sohasem fordult elő. Talán valami hiba van a programban..." Való igaz, hogy a számítógépek korántsem világmesterek még, de elnézni, „tudatosan" anyagi hátrányba kerülni – mivel minden lehetséges hadállás elemeit, elsősorban a bábok anyagi erejét pontokban értékelik, és matematikai alapon választják ki az esedékes lépést – szinte képtelenek. A játszmában egyébként *36. – Bxd5 37. Vb8+, Kf7 38. h3* következett, és sötét nyerésre állt. Előnyét azon-

ban nem használta ki, és a játszma döntetlen lett.

Alkalmunk volt a New York-i világbajnokság után látogatást tenni Spracklenék programalkotó műhelyében, San Diegóban. A hadállást betáplálták Prestige-programmal ellátott Apple II számítógépükbe, amelynek processzora ugyanaz, mint a sakk-számítógépé, csak négyszer olyan lassú (4 MHz-cel szemben 1 MHz), és képernyőjén követni lehet, hogy mely lépéseket elemzi, és hány ponttal értékeli a kialakuló hadállásokat. Így akarták ellenőrizni, nincs-e valami programhiba, amelyet – prototípusról lévén szó – még ki tudnának javítani. A saját szemünkkel láttuk, hogyan okoskodott, illetve bocsánat, számolt a gép. Elsőként 1. Hd3-mal próbálkozott, ami igen logikus, jónak tűnő lépés, mert – támadja a bástyát; – elzárja a „d" vonalat és így védi a mattot – támadja az alapsori matt kivédése után a c5 gyalogot; a huszár pedig persze nem üthető. De csakhamar „rájött", hogy az egyszerű Bd8 válaszra lóg a huszár, amely csak b2-re léphet, és akkor Vxc2-vel sötét döntő előnybe kerül. Ezek után hasonló meggondolásokból (vonalnyitás vezérnek, zárás a sötét bástyának) Hd5-re tér rá. Jelentékeny számú, de az előbbinél kevesebb mínusz pontszámot mutatott ki a Bxd5 után kialakuló pozícióra. Ezután Ba1-et elemezte, de Vxc2 után megint vesztettnek értékelte helyzetét, ezért visszatért Hd5-re, amit – amint említettük – meg is lépett. Több órán keresztül hagyták Spracklenék „gondolkodni" a gépet, amely csak a Prestige gyorsabb működési idejére átszámítva is túlságosan, a versenyen megszabott, átlag három percnyihez képest gyakorlatilag elviselhetetlenül hosszú idő után állapította meg végleg, hogy a viszonylag legkevésbé rossz lépés h3, ami igaz is. De így is lényegesen hátrányosabb az állása. A végső konklúzió: nincs programhiba; a számítógép, noha tisztet vesztett, a viszonylag esélyesebb lépések egyikét húzta meg. Ez az eset érdekes lehet mindenki számára, aki sakkprogramozással kíván foglalkozni.

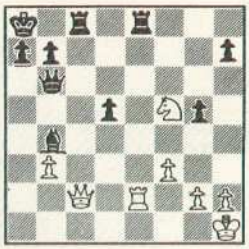
Teszt a tévé nyilvánossága előtt

A televízió A Hét műsorának riportere, az Új Tükör rovatvezetője, Molnár Károly azt javasolta, adjunk fel négy számítógépnek és négy embernek egy érdekes hadállást, és a műsor nézőinek mutassuk be, melyik készülék, illetve szereplő mit lép. A következő állást választottuk:



A világbajnok Elite A/S mikroszámítógép

Teschner – Portisch
Monaco, 1969



Sötét lép

Nagymesterünk 1. – Va6-ot lépett, és a játszma döntetlen lett. Kérdésünk az volt: mely húzással nyerhette volna meg sötét a játszmát? Amikor az állást a tesztben résztvevő négy programozónak, Kathe Spracklennek, David Kittingernek, Thomas Nitschének és David Levynek megmutattuk, valamennyinek az volt a véleménye: készüléküknek a megadott 3 percen belül megtalálja a nyerő, szép *1. – Vf2!!* lépést (a vezér ütésére, akár rögtön, akár 2. Vxc8 +, Bxc8 után Be1, illetve Bc1 matt; de védekezni sem lehet a matt és Vxc2 ellen), kivéve Thomas Nitschét, aki azt mondta: a Mephisto program annyira szelektív, egy-egy plauzibilis lépésen oly soká gondolkodik, hogy a kötött időn belül aligha kerül sor erre a valószínűtlen lépésre. Így is történt. Prestige 52 mp alatt, Chess 2001 is 3 percen belül meghúzta Vf2-t, Novag Robotjának (amely nem vett részt a versenyen, de az igazgató, Peter Auge elhozta, bemutatni a közönségnek), valamivel több, mint 3 perc (utóbb Kittinger Super Constellationnek is feladta: csupán 19 mp-re volt szüksége!), és Mephistonak valóban 5 és fél perc kellett; addig egyfolytában Portisch lépését, Va6-ot elemezte, nyilván igen nagy mélységig. Ezt a műsort az egész ország látta (magunk is nagy érdeklődéssel néztük a versenyteremben). Úgy hisszük, sokak számára érdekes lehet e „kulisszatitok" feltárása, amely ha nem is perdöntő, de jellemző adalék az „A vagy B stratégia" vitához.

Feladványfejtő teszt

A VB egyik kísérő rendezvénye volt az a teszt, amelynek során három számítógép – Mark VI., Prestige és Elite – és három fejtőprogram – szerzőik a finn Mika Korhonen és Ilkka Blom, valamint Szálka Imre – két-két 2, 3 és 4 lépéses feladványt fejtett meg. Azért választottuk az említett három számítógépet, mert ismereteink szerint csak ez a három képes esetleges több megfejtés kimutatására. A következő feladványnak két megoldása van, s valóban, mind a hat résztvevő mindkettőt megtalálta: *1. Hg7! és 1. Hd4!*

Benedek Attila

Matt 2 lépésben

Ez a feladvány jó példa arra, hogy miért fontos a számítógépnek ez a képessége, hiszen a szerző intenciója itt két megfejtés; mindkét lépés azonos stratégiájú, felold egy-egy sötét bábot, amely a világos királynak két sakkot adhat. A számítógépek és programok fejtési ideje a következő volt: Mark VI. 17, Prestige 7, Elite 11, Korhonen 3, Blom 14, Szálka 5 mp. A többi feladványnál a fejtési idők természetesen másképpen alakultak – a négylépéseseknél 300 és 800 mp között jártak. Összesítve: a számítógépek közül Mark VI., a programok közül Korhonené volt a leggyorsabb. Az összehasonlítás – amiről külön tanulmány készült – a programok algoritmusával kapcsolatban igen érdekes következtetésekre nyújt lehetőséget.

DR. LINDNER LÁSZLÓ

MTA SZTAKI SYSTER–VARYTER – mikroszámítógépek

Jelen évtizedben a mikroszámítógépek széles körű elterjedésének vagyunk tanúi. Ezek a berendezések a nem számítástechnikával foglalkozó szakemberek számára is mindennapi segédeszközzé válnak.
Már hazánkban is igen nagy igény mutatkozik feladat-orientált mikroszámítógép-rendszerek használatára. Adott alkalmazások különleges igényeinek számítástechnikai eszközökkel történő kielégítése napi problémáink közé került. Az emberek nem tekintik idegennek az ilyen berendezéseket.
A Magyar Tudományos Akadémia Számítástechnikai és Automatizálási Kutató Intézetében a számítógéphálózat-fejlesztési munka során létrejött SYSTER és VARYTER típusú alkalmazói terminálok mikroszámítógépek. Alkalmasak a legkülönbözőbb területeken való felhasználásra, például:

irodai ügyvitelben,
nyomdai szövegfeldolgozásban,
egészségügyi nyilvántartó és diagnosztikai rendszerekben,
mezőgazdasági termelésirányításban,
sportban, eredmény nyilvántartására,
geodéziai mérések kiértékelésére stb.

A SYSTER mikroszámítógép jellemzői
A SYSTER alkalmazás-orientált mikroszámítógép adott célfeladatok megoldását teszi lehetővé. Kezelése nem igényel számítástechnikai ismereteket. Az ember–gép kapcsolatot párbeszédes üzemmódban valósítja meg. A célfeladat megoldásához esetleg szükséges számítógép-háttér használatát terminál üzemmódban biztosítja.

Felépítése
8 bites mikroprocesszor köré épített mikrogép. Központi egysége egykártyás kivitelű. Perifériavezérlő készlete bővíthető. A billentyűzet és a képernyő a mikrogép szerves része.

Központi egység
A központi egység Z80–A típusú mikroprocesszort tartalmaz (ciklusideje 2,5 μs). Az inicializáló és célprogramok tárolására maximálisan 16 kbájt kapacitású csak olvasható tár (ROM) szolgál. Betöltött programok tárolása a maximálisan 128 kbájt kapacitású író/olvasható tárban (RAM) történik. A központi egység kártya tartalmazza még a képernyővezérlő, klaviatúra-illesztő, hajlékony mágneslemez tároló vezérlő áramköröket. A nyomtató berendezés csatlakoztatása párhuzamos interfészen keresztül történik, a normál kazettás magnetofon V24-es interfészen keresztül csatlakoztatható.

Képernyő
Az alkalmazott képernyő normál, kereskedelemben közvetlenül beszerezhető fekete-fehér, 15 cm vagy 31 cm átmérőjű tv-készülék. A készülék monitor üzemmódban használható, video bemenetén csatlakozik a központi egységre.

Billentyűzet
A klaviatúra hall generátoros billentyűkből épül fel. Elrendezése maximálisan 7 × 32 rasztermezőben történhet. A billentyűelrendezést a felhasználó meghatározhatja, szabadon programozhatja. A billentyű megnyomásakor hangvisszajelzés történik, nyolcféle hangmagasság lehetséges.

Illeszthető perifériák
A mikrogéphez kapcsolható perifériák
hajlékony mágneslemezes tároló
(maximálisan 4 db meghajtó egység; szimpla-dupla írássűrűség; egy-, kétoldalas hajlékony mágneslemezek)
kompakt kazettás magnetofon
nyomtató (mátrix, sor, margarétakerekes)
lyukszalag olvasó/lyukasztó
vonalkódolvasó
speciális perifériák (jegy, számla, nyomtató)
speciális mérőműszerek, adatgyűjtők stb.
A perifériák illesztése a központi egységhez a perifériavezérlőkön keresztül történik. Közülük a speciális illesztők csak egy adott periféria illesztését végzik (például a hajlékony mágneslemezes tároló vezérlő), de lehetséges a csatlakoztatás BSI jellegű párhuzamos interfészen vagy V24 soros interfészen keresztül is. Természetesen a vezérlési szekvenciák programozhatók.

A SYSTER alkalmazás-orientált mikroszámítógépeknek két változata létezik:

SYSTER-MODELL A
SYSTER-MODELL B

A MODELL–A képernyője 15 cm-es, a MODELL–B-é 31 cm-es. Az előbbinél a szöveg 80 × 200-as raszterpont-mezőben grafikusan és alfanumerikusan, az utóbbinál 25 sornyi, soronként 80 karakteres megjelenítés lehetséges.

A SYSTER MODELL-A alkalmazási változatai

Nyomdai szövegrögzítő. A SYSTER mikrogép ebben a változatában tetszőleges szövegek sorfolytonos rögzítését végezheti adott adathordozóra (például lyukszalagra, hajlékony mágneslemezre, kompakt mágnesszalagra stb.). A képernyő két sora a szedés paramétereinek megjelenítésére szolgál, a további négy sor pedig a rögzíteni kívánt szöveget tartalmazza. A grafikus ernyő gyorsan váltható, nyomdai igényt kielégítő karaktermegjelenítést tesz lehetővé. A billentyűzet tetszőleges elrendezésű és kódolású lehet, a nyomdai hagyományokhoz való alkalmazkodás céljából.

Jegykiadó terminál (fejlesztés alatt). Rendeltetése valamilyen központi gép felügyelete mellett raktári, közlekedési, színházi, sportrendezvényi jegyek lekérdezése, kiadása. A grafikus képernyő alkalmas a jegy teljes vagy egy-egy részletének megjelenítésére. A szükséges adatok lehívása és bevitele után a jegynyomtató elkészíti a jegyet. A klaviatúra főleg funkcionális billentyűket tartalmaz, ami a szükséges billentyűzések számát nagyon lecsökkenti, elrendezése teljesen alkalmazkodik a felhasználóhoz. Bővítésnél vagy önálló jegykiadás esetén háttértárolóval bővíthető, és a soros csatolás esetleg feleslegessé válik.

Pénztári terminál (fejlesztés alatt). Feladata vagy kézi bevitellel vagy vonalkód (Bar kód) alapján a vásárolt áruk árainak összegzése, kijelzése és központba küldése, illetve a visszajáró pénz kiszámolása.

A grafikus képernyő alkalmas nagyméretű számok, valamint részletösszegek megjelenítésére. A billentyűzet a hagyományos pénztárgépek elrendezését követi. A csíkolvasó ceruza adatát a program kiértékeli, a blokkot a jegynyomtató elkészíti, az összeget pedig a központi gépnek megküldi.

Egyedül álló üzleteknél a napi forgalom rögzítéséhez háttértárolóra lehet szükség.

Játék-számítógép (fejlesztés alatt). Feladata a legegyszerűbb nyelvek interpretálása pl. BASIC, assembler, melyet gyerekek is elsajátíthatnak, illetve játékprogramokat futtathatnak. A grafikus ernyő alkalmas 10 sorban 32 karakter megjelenítésére programíráshoz, valamint szép ábrák rajzolására a játékok számára. A klaviatúra elrendezése a szükséges ASCII karakter mellett a játékokhoz szükséges vezérlő karaktereket is tartalmazza. Az elkészített programok a kazettára írhatók és visszahívhatók.

A SYSTER MODELL–B mikrogép alkalmazási változatai

Számlázó automata. Feladata számlák készítése, cikklisták alapján. Az elkészült számla kinyomtatása. Napi eredmények közlése a központi géppel.

Raktári nyilvántartó
Feladata a szokásos kartonrendszer kiváltása. Bevétel–kiadás, ekvivalens anyagok ismerete, a napi zárás megküldése a központnak, listák készítése különböző szempontok szerint.

Irodai szövegszerkesztő
Feladata levelek iktatása, írása; címlisták, napi teendők jegyzése; nevek, címek visszakeresése. Levél minőségű a szöveg megjelenítése nyomtatón.

Személyi számítógép
Személyi felhasználású számítógép. Feladata programok írása, szerkesztése, futtatása.

A VARYTER mikroszámítógépet a következő számban mutatjuk be.

az MTA SZTAKI mikro-számító-gépei

ÁRA: 28,– FORINT

NOVOTRADE RT

H–1136 BUDAPEST,
FÜRST SÁNDOR U. 24–26.
TELEFON: 530-022
TELEX: 22-5959 NOVTR H